The Dissertation Committee for Michael Glen Anderson
Certifies that this is the approved version of the following dissertation:

Design of Multiple Frequency Continuous Wave Radar Hardware and Micro-Doppler Based Detection and Classification Algorithms

Committee:

Hao Ling, Supervisor

Robert Rogers, Co-supervisor

Francis Bostick

Dean Neikirk

Jeffrey Andrews

Steven Poole

Design of Multiple Frequency Continuous Wave Radar Hardware and Micro-Doppler Based Detection and Classification Algorithms

by

Michael Glen Anderson, M.S.E.; B.S.E.E.

Dissertation

Presented to the Faculty of the Graduate School of
The University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

The University of Texas at Austin
May 2008

Dedication

To my parents Alfrieda Helen Anderson and Glen Melvin Anderson, who planted the seed of knowledge in me from day one and have lovingly nurtured its growth ever since. My parents taught me lessons on life that were far more profound than anything that I have ever learned in a classroom or through my research. And to my wife Jessica Lee Anderson, who inspires me to follow my dreams and to be a better person. Jessica helped me throughout the dissertation process, providing me encouragement, an always sympathetic ear, better-than-gourmet meals, and more than her fair share of work around the house. To top it all off, Jessica literally crawled on her hands and knees to help me finish my dissertation. I am forever grateful for Jessica's love, passion, and friendship.

Acknowledgements

I would like to express my sincerest thanks to my supervisors, Professor Hao Ling and Dr. Robert Rogers. Their invaluable academic and professional assistance will affect my career for years to come. I also greatly appreciate the service of my committee members who have graciously taken time out of their busy schedules.

Most of the research for this dissertation occurred within Applied Research Laboratories (ARL) at The University of Texas at Austin. I am grateful to have utilized such state-of-the-art facilities for my research. I want to thank several members (and former members) of the ARL staff for their assistance with data collection including: Alfred “AJ” Walker, Kuan Li, William “Mac” Junod, Franklin “Frankie” Fitzhugh, Glen Olinger, Jessica Nixon, and George Monnat. This work was supported by the Defense Threat Reduction Agency (DTRA) and Applied Research Laboratories internal research and development funds.

I want to express my appreciation to Kevin Murphy for use of the HMM toolbox [1]. I would also like to thank Chih-Chung Chang and Chih-Jen Lin of National Taiwan University for use of their LIBSVM software [2].

Mrs. Janelle Holman and Mrs. Caroline Erlinger deserve immense thanks for financially supporting me as a graduate student through the Thrust 2000 Archie W. Straiton Endowed Graduate Fellowship in Engineering. I would also like to thank all of the members of the ARL Educational Programs staff.

Finally, I wish to thank all of my family and friends for their support, love, patience, and understanding.

Design of Multiple Frequency Continuous Wave Radar Hardware and Micro-Doppler Based Detection and Classification Algorithms

Publication No._____________

Michael Glen Anderson, Ph.D.
The University of Texas at Austin, 2008

Supervisor: Hao Ling
Co-supervisor: Robert Rogers

Micro-Doppler is defined as scattering produced by non-rigid-body motion. This dissertation involves the design of a multiple frequency continuous wave (MFCW) radar for micro-Doppler research and detection and classification algorithm design.

First, sensor hardware is developed and tested. Various design tradeoffs are considered, with the application of micro-Doppler based detection and classification in mind. A diverse database of MFCW radar micro-Doppler signatures was collected for this dissertation. The micro-Doppler signature database includes experimental data from human, vehicle, and animal targets. Signatures are acquired from targets with varying ranges, velocities, approach angles, and postures. The database is analyzed for micro-Doppler content with a focus on its application to target classification.

Joint time-frequency detection algorithms are developed to improve detection performance by exploiting noise-spreading and the micro-Doppler phenomenon. Following detection algorithm development, this dissertation covers the design of micro-

Doppler feature extraction, feature selection, and classification algorithms. Feature selection is performed automatically via a Fisher score initialized sequential backward selection algorithm. Classification is performed using two distinct approaches: a generative statistical classification algorithm based on Gaussian mixture models (GMMs) and a discriminative statistical classification algorithm based on support vector machines (SVMs).

Classifier performance is analyzed in detail on a micro-Doppler signature database acquired over a three-year period. Both the SVM and GMM classifiers perform well on the radar target classification task (high accuracy, low nuisance alarm probability, high F-measure, etc.). The performance of both classifiers is remarkably similar, and neither algorithm dominates the other in any performance metric when using the chosen feature set. (However, the difference between SVM and GMM classification accuracy becomes statistically significant when many redundant features are present in the feature set.) The accuracy of both classifiers is shown to vary as a function of approach angle, which physically corresponds to the angular dependence of micro-Doppler. The results suggest that overall classifier performance is more sensitive to feature selection than classifier selection (with GMM being more sensitive to redundant features than SVM). Both classifiers are robust enough to handle human targets attempting to evade detection by either army crawling or hands-and-knees crawling.

Table of Contents

Table of Contents viii

List of Tables xii

List of Figures xiv

Chapter 1: Introduction 1
    1.1 Motivation 1
    1.2 Problem Statement 1
    1.3 Complexity of Non-cooperative Target Recognition 2
    1.4 Approach 2
    1.5 Summary of Objectives 4
    1.6 Related Work 4
    1.7 Dissertation Contributions 6
    1.8 Organization of the Dissertation 8

Chapter 2: Joint Time-Frequency Signal Processing 10
    2.1 Motivation for Joint Time-Frequency Processing 10
    2.2 Introduction to Joint Time-Frequency Processing 11
    2.3 The Short-Time Fourier Transform 12

Chapter 3: The Micro-Doppler Phenomenon 15
    3.1 Instantaneous Frequency 15
    3.2 Point Scatterer Model and Example Spectrogram 18

Chapter 4: Sensor Selection and the MFCW Radar 20
    4.1 Sensor Requirements 20
    4.2 Choice of Sensor 21
    4.3 Multiple Frequency Continuous Wave Radar Architecture 24
    4.4 MFCW Radar Target Range Calculation 27

Chapter 5: Target Detection and Ranging 32

5.1 Radar Range Equation 32

5.2 Classical Detection Theory 33

5.3 Detection in the Joint Time-Frequency Domain 38

5.4 Procedure for Joint Time-Frequency Domain Detection 42

5.5 Ranging in the Joint Time-Frequency Domain: The Time-Frequency-Range Diagram (TFRgram) 45

Chapter 6: Radar Hardware Design 49

6.1 Radar System-Level Design 49

6.2 MFCW Radar Design 50

6.3 System Design for Reduced Noise Figure 52

6.4 Radar Parts Cost Analysis 54

Chapter 7: Database Collection and Pre-processing 56

7.1 Data Collection Environments 56

7.2 Database Collection 60

7.3 Database Pre-Processing 61

Chapter 8: Detection Performance and Micro-Doppler Analysis 64

8.1 Detection Performance 64

8.2 Micro-Doppler Analysis 68

Chapter 9: Classifier Design and Statistical Analysis 80

9.1 Introduction 80

9.2 Statistical Feature Extraction 82

9.3 Heuristic Feature Extraction 86

9.4 Feature Selection 89

9.5 The Confusion Matrix and Classifier Performance Metrics 94

9.6 Population Parameter Estimation Using Sample Statistics 97

9.7 Distribution of the Sample Mean Estimator When σ_X is Known 100

9.8 Distribution of the Sample Mean Estimator When σ_X is Unknown ...101

9.9 Population Mean Confidence Interval Estimators 103

9.10 Statistical Significance Testing for Classifier Comparison 104

Chapter 10: Support Vector Machines..109
10.1 Introduction to Support Vector Machines..109
10.2 Non-Linear Support Vector Machines for Non-separable Training Data 115
10.3 Practical Considerations for Using Support Vector Machines 117

Chapter 11: Gaussian Mixture Models 120
11.1 Sequence Classification: The MAP and ML Decision Rules 120
11.2 Mixture Modeling 124
11.3 Gaussian Mixture Model Classification...129

Chapter 12: Classifier Training..131
12.1 Training, Parameter Tuning, and Testing Datasets............................131
12.2 Stratified Database Partitioning 132
12.3 SVM Performance Versus the Cost and Gamma Parameters............136
12.4 Number of Support Vectors Versus the Cost Parameter 137
12.5 GMM Performance Versus the Number of Gaussian Mixtures.........139
12.6 Performance of Selected Classifiers 140
12.7 Classifier Performance Using Only Statistical Features....................146
12.8 Classifier Performance Using Only Heuristic Features.....................147
12.9 Classifier Performance Without the Use of Range Information.........148
12.10 Classifier Performance Using Fisher Score Feature Set..................150
12.11 Statistical Significance Testing Using Cross-Validation.................152

Chapter 13: Classifier Performance Evaluation.....................................157
13.1 Introduction...157
13.2 Classification Results...157
13.3 Confidence Intervals and Statistical Significance Testing.................164
13.4 Analysis of Classification Results 166

Chapter 14: Conclusion..168
14.1 Dissertation Contributions 168
14.2 Future Work 170

References..171

Vita……………………………………………………………………...177

List of Tables

Table 4.1. Sensor system requirements 20
Table 4.2. Ranging parameters 31
Table 5.1. Detection parameters 41
Table 6.1. Factors influencing radar parameter selection 50
Table 6.2. Front-end component noise figures and gains 53
Table 6.3. Varying receiver noise figure by changing number and position of LNAs 54
Table 6.4. Small quantity radar parts cost analysis 55
Table 8.1. Maximum detectable target ranges for $P_d = 0.9$ and $P_{fa} = 3.3\times10^{-7}$ 65
Table 8.2. Empirical maximum detectable range 67
Table 9.1. FSISBS selected classification features. 94
Table 9.2. Confusion matrix example 1 94
Table 9.3. Labeled confusion matrix 95
Table 9.4. Confusion matrix example 2 96
Table 11.1. Total number of GMM free parameters 129
Table 12.1. Stratified database partitioning at the class level 133
Table 12.2. Stratified database partitioning of the human class 134
Table 12.3. Stratified database partitioning of the vehicle class 135
Table 12.4. Stratified database partitioning of the animal class 136
Table 12.5. Grid-search over SVM cost parameter C and kernel parameter γ. Performance is characterized by F-measure and accuracy (in parenthesis). 137
Table 12.6. SVM performance using FSISBS feature set 141
Table 12.7. SVM performance on human sub-classes 142
Table 12.8. SVM performance on vehicle sub-classes 142
Table 12.9. SVM performance on animal sub-classes 142
Table 12.10. GMM performance using FSISBS feature set 144
Table 12.11. GMM performance on human sub-classes 145
Table 12.12. GMM performance on vehicle sub-classes 145
Table 12.13. GMM performance on animal sub-classes 145
Table 12.14. SVM performance using only statistical features 147
Table 12.15. GMM performance using only statistical features 147
Table 12.16. SVM performance using only heuristic features 148
Table 12.17. GMM performance using only heuristic features 148
Table 12.18. SVM performance without the use of range information 149
Table 12.19. GMM performance without the use of range information 150
Table 12.20. SVM performance using feature set selected by Fisher score only 151
Table 12.21. GMM performance using feature set selected by Fisher score only 151
Table 12.21. Sample statistics, confidence intervals, and paired Student's t-test using FSISBS feature set 154

Table 12.22. Sample statistics, confidence intervals, and paired Student’s t-test using Fisher score feature set 156
Table 13.1. SVM performance on testing set 158
Table 13.2. SVM performance on human sub-classes (testing set) 160
Table 13.3. SVM performance on vehicle sub-classes (testing set) 160
Table 13.4. SVM performance on animal sub-classes (testing set) 160
Table 13.5. GMM performance on testing set 161
Table 13.6. GMM performance on human sub-classes (testing set) 163
Table 13.7. GMM performance on vehicle sub-classes (testing set) 163
Table 13.8. GMM performance on animal sub-classes (testing set) 163
Table 13.9. Testing set sample statistics, confidence intervals, and paired Student’s t-test 165

List of Figures

Figure 2.1. Simulated time series (a) and DFT (b) of three-frequency waveform 11
Figure 2.2. Spectrogram of stepped-frequency waveform... 14
Figure 3.1. Radar configuration 15
Figure 3.2. Constant velocity (a), constant acceleration (b), and constant jerk (c) 17
Figure 3.3. Human point scatterer model with five scattering centers 18
Figure 3.4. Micro-Doppler features of a running human... 19
Figure 4.1. MFCW radar block diagram.. 24
Figure 4.2. Quadrature demodulator 26
Figure 4.3. Three tone frequency separation 30
Figure 5.1. Probability density functions of noise and signal-plus-noise 35
Figure 5.2. Probability of detection versus SNR (a), and receiver operating curve (b).... 37
Figure 5.3. Complete spectrogram (a), extracted frame (b), and detected frame (c)........ 43
Figure 5.4. Spectrogram frame for human target (a), detection results for threshold level: -55dBm (b), -61dBm (c), and -76dBm (d)... 44
Figure 5.5. Spectrogram (a) and TFRgram (b) of human jogging from 292ft to 15ft...... 46
Figure 5.6. TFRgram after application of a threshold mask 47
Figure 5.7. Centroid frequency (a) and target range (b) 48
Figure 6.1. Receiver front end with two LNAs before the power dividers 53
Figure 7.1. Industrial testing environment in Austin, Texas.. 56
Figure 7.2. Ranch in central Texas where goat datasets were collected.......................... 57
Figure 7.3. Deer data collection site in an Austin, Texas suburban neighborhood. 58
Figure 7.4. Bird data collection site in an Austin, Texas suburban neighborhood........... 59
Figure 7.5. Mobile data collection platform (a), and close-up view (b) 59
Figure 7.6. 3FCW radar pre-processing prior to joint time-frequency detection 61
Figure 7.7. Magnitude of fourth-order elliptic high-pass filter transfer function 62
Figure 7.8. Spectrogram with strong image target (a), and corrected spectrogram (b) 63
Figure 8.1. Spectrogram of jogging human before (a), and after (b) detection 66
Figure 8.2. Number of detections per FFT (a), and received power (b) for jogging human 67
Figure 8.3. Spectrogram of human running (a), jogging (c), and walking (e). Expanded views are (b), (d), and (f), respectively.. 69
Figure 8.4. Human target walking at 45º (a) and 90º (b) relative to radar........................ 70
Figure 8.5. Human crawling (a). Transition from walking to crawling (b). 71
Figure 8.6. Vehicle driving at different velocities and accelerations 72
Figure 8.7. Vehicle driving at 45º (a) and 90º (b) relative to radar.................................. 73
Figure 8.8. Vehicle turning in front of radar at close range... 74
Figure 8.9. Micro-Doppler produced by vehicle engine is visible at close range 75
Figure 8.10. Spectrogram of dog (a), goat (c), and deer (e). Expanded views are (b), (d), and (f), respectively 76
Figure 8.11. Spectrogram of bird (a) and expanded view (b).. 77

Figure 8.12. Multiple targets moving in opposite directions (a), and the same direction (b) 78
Figure 8.13. Spectrogram of ambient conditions (a), and human target (b) during heavy rain 79
Figure 9.1. Block diagram of a typical real-time classification system 81
Figure 9.2. LPCC feature extraction 84
Figure 9.4. Calculating micro-Doppler period: spectrogram (a), micro-Doppler energy $E_{micro-Doppler}[n]$ (b), and autocovariance (c) 88
Figure 9.5. Fisher score of all 80 extracted features 93
Figure 9.6. Student's t-distribution with 9 degrees of freedom 102
Figure 9.7. The Student's t-test ($t_9 = 1.5$) $p-$value is the area of the shaded region 107
Figure 10.1. Support vector machines 109
Figure 10.2. Adding slack variables to avoid over-fitting training data 111
Figure 11.1. Gaussian mixture model sequence classification. The ML decision rule is implemented when the switches are open. Closing the switches implements the MAP decision rule. 130
Figure 12.1. Number of support vectors versus C for human-versus-rest (a), vehicle-versus-rest (b), and animal-versus-rest (c) SVMs. Total number of support vectors in (d). 138
Figure 12.2. GMM performance versus the number of Gaussian mixtures M 140

Chapter 1: Introduction

1.1 MOTIVATION

Sensors monitor their surroundings and facilitate the detection of objects of interest. False alarms occur when a sensor incorrectly declares that system noise is the object of interest. Classical Neyman-Pearson detection and (more recently) joint time-frequency detection provide methods for controlling the false alarm rate. Practical sensors must also minimize the number of *nuisance alarms*. Nuisance alarms are false detections triggered by objects other than the object of interest (and *not* caused by system noise). Fielded sensors may exhibit robust performance against false alarms, but these sensors are often unequipped with the classification algorithms required to minimize nuisance alarms. A high nuisance alarm rate may render an otherwise superior sensor nearly useless. High probability of detection, low false alarm rate, and low nuisance alarm rate sensors are widely applicable in both the commercial sector (asset protection, airport security, process automation, wildlife monitoring) and the military sector (perimeter monitoring, homeland security, force protection).

1.2 PROBLEM STATEMENT

The primary focus of this dissertation is the detection and classification of moving objects by a sensor on a stationary platform. The goal is to perform robust non-cooperative automatic target detection and classification. (Note that the term "target" refers to an object of interest and not a target of munitions). Detection and classification must be fully automatic, requiring no human operator interaction. The classification algorithms must be able to classify human, animal, and vehicle targets to a high degree of accuracy. The system must adequately cope with deliberate attempts to spoof the classification algorithms. The sensor should be designed to be a short to medium range sensor to facilitate improved portability and significant reductions of cost, power, and complexity as compared to typical long range recognition systems (e.g., high range resolution radar).

1.3 COMPLEXITY OF NON-COOPERATIVE TARGET RECOGNITION

Non-cooperative target recognition (NCTR) is a complex recognition task. One of the primary complexities of NCTR is the unconstrained nature of target motion. Targets undergo widely varying velocities, approach angles, dwell times, gaits, and postures. This variation directly results in increased feature spread which reduces the inherent separability of the classes. Another difficulty of NCTR is large intra-class variability. For example, the animal class may contain signatures from numerous species of animal such as deer, bird, goat, and dog. The differing size and locomotion techniques of each species result in distinct radar signatures. Sub-class target recognition may be attempted, but fine-grain identification often requires a prohibitive amount of training data. Wide ranges of received signal to noise ratio (SNR) also contribute to the complexity of NCTR. The training data must properly represent the wide variability inherent in NCTR for adequate system performance in deployment conditions.

1.4 APPROACH

The first step toward achieving the primary dissertation objective was to design a sensor system to collect target signatures for algorithm development. The chosen sensor was the multiple frequency continuous wave (MFCW) radar [3], [4]. MFCW radars simultaneously provide target range estimates and the highest possible Doppler resolution (the MFCW radar will be discussed in greater detail in Chapter 4). The high Doppler resolution of the MFCW radar enables the extraction of micro-Doppler features. Micro-Doppler refers to Doppler scattering returns produced by non-rigid-body motion. Micro-Doppler produces many detailed radar image features in addition to those associated with bulk target motion. Targets of different classes often create micro-Doppler images distinguishable even by non-expert observers. Micro-Doppler features can be implemented in new target classification systems or used to enhance existing systems with Doppler-sensitive hardware.

Following the sensor design, a diverse corpus of target signatures was collected, processed, and analyzed. The collected radar signatures contain data from humans, vehicles, and animals. The data includes novel micro-Doppler signatures such as crawling humans and a variety of animal species. Micro-Doppler was recorded from targets approaching the radar at various velocities, angles, ranges, maneuvers, and postures. The data offered valuable insight into the micro-Doppler phenomenon.

Analysis of micro-Doppler allowed for the subsequent creation of high-performance classification features based on target micro-Doppler response. The cepstral coefficients, cepstral temporal derivatives, and cepstral energy features were found to efficiently represent micro-Doppler data. In addition, the micro-Doppler energy, range-weighted target energy, and micro-Doppler period features were also selected by the automatic feature selection algorithm.

Classification was performed using the statistical pattern recognition paradigm. Classifier performance comparisons included both discriminative (decision region based) and generative (probabilistic model based) classifiers. The efficacy of support vector machines (SVMs) and Gaussian mixture models (GMMs) was analyzed on a micro-Doppler signature database collected over a three-year period.

1.5 SUMMARY OF OBJECTIVES

The following list summarizes the main research objectives:

1. Perform robust, automatic detection and classification of moving non-cooperative targets using a sensor on a stationary platform (primary objective).
2. Design low-cost, portable sensor hardware to support detection and classification algorithm development.
3. Collect, process, and analyze a diverse corpus of target signatures.
4. Research the micro-Doppler phenomenon by analyzing experimental data.
5. Design target detection and classification algorithms based on novel micro-Doppler based feature sets.
6. Perform a detailed comparative performance analysis of support vector machine (SVM) and Gaussian mixture model (GMM) classifiers.

1.6 RELATED WORK

This section reviews the literature related to this dissertation research. Research of micro-Doppler feature modeling, analysis, and classification has increased significantly over the last decade. Early micro-Doppler research focused on developing accurate mathematical models of the micro-Doppler phenomenon [5]. Various micro-Doppler models have been developed to analyze and predict the human micro-Doppler response. Some of the more complex human micro-Doppler models are found in [6] and

[7]. Micro-Doppler features have great potential for use in automatic target classification algorithms. Extraction of micro-Doppler features is typically performed in the joint time-frequency domain. The authors of [8] and [9] use chirplet techniques to perform feature extraction. A similar technique of linear FM basis decomposition is found in [10]. In [11], the author employs independent component analysis (ICA) to extract independent basis functions from the spectrogram to be used as features in a target classifier. A combination of wavelet transforms and time-frequency analysis is used in [12] to isolate individual micro-Doppler features.

Although the potential benefit of using micro-Doppler in classification algorithms is high, relatively little experimental (non-synthetic) micro-Doppler data exists. Much of the experimental micro-Doppler data comes from human subjects as in [6], [13], [14], [15], [16]. The authors of [15] extracted micro-Doppler signatures and direction-of-arrival (DOA) estimates at over 9 meters range through a brick wall. Fully polarimetric human radar signatures at different speeds and approach angles with respect to the radar were collected by the authors of [16]. In [17], [18], and [19], automobile micro-Doppler signatures are presented. Field data from a helicopter is shown in [14] and [20]. Micro-Doppler data from animals is quite limited in the literature. The authors of [21] collected horse, cow, dog, sheep, and pig micro-Doppler data using a ground surveillance pulse-Doppler radar. The author of [22] collected dog micro-Doppler signatures using a simple CW radar.

Micro-Doppler features have been utilized in target classification algorithms in [13], [14], [19], [21], [22], [23], [24], and [25]. In [13], the authors attempt to classify individual humans using their micro-Doppler signatures as a biometric. Feature vectors based on spectrogram averages were used to distinguish individual humans. The authors of [14] performed classification among human, wheeled vehicles, tracked vehicles, and helicopters. The classifier utilized a combination of linear discriminant analysis (LDA) for dimensionality reduction and a Bayesian classifier. In [19], data collected from a low-cost 24 GHz CW radar is used to classify vehicles into three groups: cars and sports utility vehicles, vans, and buses and trucks. The authors extracted classification features from the spectrogram with a combination of the Hough transform and principal components analysis (PCA).

The authors of [21] performed automatic target classification among human, wheeled vehicle, tracked vehicle, and animal classes. The static cepstral coefficients were utilized as features in Gaussian mixture model (GMM) classifiers. In [22], the author utilized a heuristic feature set to classify CW radar data into single male targets, single female targets, and groups of human targets. Target velocity, stride length, and a ratio of appendage to torso radar cross section (RCS) were the features incorporated into the classifier. The authors of [23] used dynamic time warping (a template matching algorithm) to classify humans, tracked vehicles, and wheeled vehicles. The template matching approach was a unique feature of this work. A simulation study was performed in [24] where the authors classify four types of micro-Doppler features: vibration, rotation, coning, and tumbling. Linear discriminant analysis (LDA), k-nearest neighbors (K-NN), and support vector machine (SVM) classifiers were compared on the same dataset. In [25], the authors used hidden Markov models (HMMs) to classify human, tracked vehicle, and wheeled vehicle targets. The HMM classifier utilized LDA-compressed spectrograms as feature vectors.

1.7 DISSERTATION CONTRIBUTIONS

This dissertation has made the following contributions:

Collection, processing, and analysis of a MFCW radar micro-Doppler signature corpus:

A diverse corpus of MFCW radar signatures was collected, processed, and analyzed for this dissertation. The data was collected in realistic operational environments including an industrial building complex, a suburban neighborhood, and a small ranch. The corpus includes datasets with targets moving at various ranges, approach angles, and velocities with respect to the radar. Humans, vehicles, and animals were the main target classes. Novel micro-Doppler signatures were collected, including human crawlers (both army crawl and hands-and-knees "baby" crawl), birds, goats, and deer. The data were processed and analyzed for micro-Doppler content.

MFCW radar hardware design and target range visualization:

This dissertation contributed the hardware design of a low-cost, high-performance multiple frequency continuous wave (MFCW) radar. The radar hardware can be used for high quality micro-Doppler signature collection in either academic or industrial settings. This dissertation also introduced the time-frequency-range diagram (TFRgram). The TFRgram is a novel method of visualizing and extracting range data from multiple frequency continuous wave (MFCW) radars.

Novel micro-Doppler based feature set:

This dissertation identified a novel set of micro-Doppler based classification features. The micro-Doppler energy, range-weighted target energy, micro-Doppler period, and the cepstral coefficients provided a feature set with high discriminatory power. The cepstral coefficients, cepstral temporal derivatives, and cepstral energy features efficiently represented the micro-Doppler content of the data. The influence of feature selection on classifier performance was studied in detail.

Design and analysis of micro-Doppler based classification algorithms:

This dissertation research involved the design and analysis of classification algorithms based on micro-Doppler features. Both support vector machine (SVM) and Gaussian mixture model (GMM) classifiers were analyzed for their performance on the radar target classification problem. Algorithms that perform well on data taken at closely spaced times may dramatically degrade when presented with data taken at much different time frames. This dissertation evaluated classifier performance on data collected over a three-year period.

1.8 ORGANIZATION OF THE DISSERTATION

Chapter 2 introduces the main concepts of joint time-frequency processing. The short-time Fourier transform (STFT) is discussed as it is used extensively in this dissertation to analyze target micro-Doppler. Chapter 3 serves as an introduction to the micro-Doppler phenomenon. Both simulated and empirical micro-Doppler signatures are included in this chapter.

Chapter 4 begins with a discussion of sensor specifications and competing sensor technologies. An investigation of the multiple frequency continuous wave (MFCW) radar architecture follows the design specifications. Chapter 4 concludes with the theory of target ranging using MFCW radars.

Chapter 5 develops the theory of MFCW radar target detection. First, important results from classical detection theory are reviewed. The following section covers joint time-frequency detection. The remaining sections develop specific procedures for target detection and ranging in the joint time-frequency domain, including an introduction to the time frequency range diagram (TFRgram).

The design of MFCW radar hardware is covered in Chapter 6. The discussion is focused on low-noise receiver design. Chapter 7 begins with an overview of data collection and a description of the contents of the radar signature database. The chapter ends with a discussion of signal processing prior to joint time-frequency detection.

The results of target detection testing and micro-Doppler signature analysis are contained in Chapter 8. The first section covers detection performance. The remaining sections display and analyze a wide variety of micro-Doppler signatures. Chapter 9 presents aspects of target classification theory relevant to this work. Feature extraction, feature selection, confusion matrices, and the theory behind classifier performance analysis are included in the discussion.

Chapters 10 and 11 cover the theory of support vector machines (SVMs) and Gaussian mixture models (GMMs), respectively. Classifier training and parameter tuning is presented in detail in Chapter 12. Chapter 12 includes a detailed study of the impact of feature selection on classifier performance.

Classification results on the testing set are presented and evaluated in Chapter 13. Chapter 14 concludes the dissertation with an overview of the dissertation contributions and a discussion of future work.

Chapter 2: Joint Time-Frequency Signal Processing

2.1 MOTIVATION FOR JOINT TIME-FREQUENCY PROCESSING

This chapter reviews important concepts from joint time-frequency processing of baseband signals. Consider the complex signal $x_{bb}(t) = i(t) + jq(t)$ where $i(t)$ is the received signal in-phase component and $q(t)$ is the quadrature component. The signal $x_{bb}(t)$ is an (approximately) band-limited continuous-time waveform containing both amplitude and phase modulation. An analog to digital converter samples the baseband signal at the rate of F_s Hz. Thus, the digitized waveform is

$$x[n] = x_{bb}(nT_s) \qquad T_s = 1/F_s \,. \tag{2.1}$$

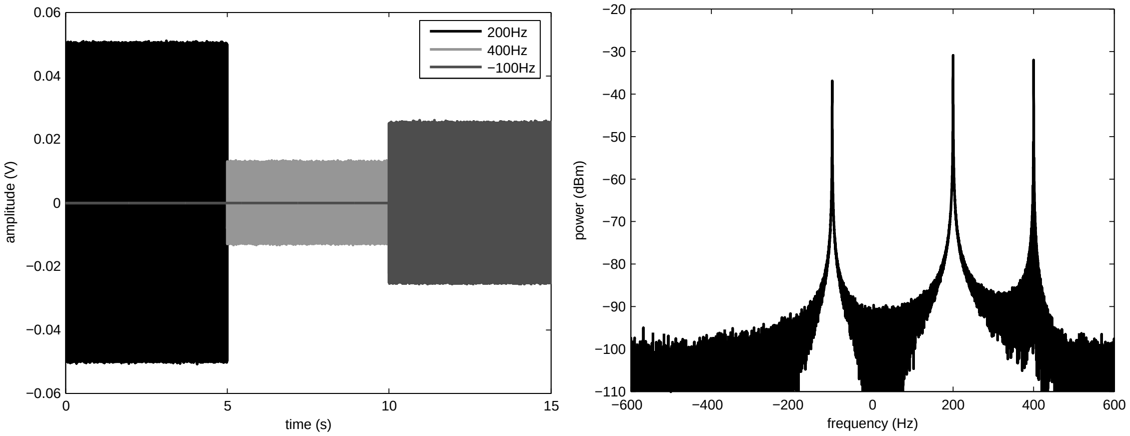
where T_s is the sampling period and n is an integer discrete time index. For stationary signals, the discrete Fourier transform (DFT) is employed to analyze spectral content. The DFT of a discrete time signal $x[n]$ is defined as

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N}, \quad k = 0, 1, \cdots, N-1 \tag{2.2}$$

where k is the discrete frequency index and N is the length of the DFT . Similarly, the inverse discrete Fourier transform (IDFT) is defined as

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j2\pi nk/N}, \quad n = 0, 1, \cdots, N-1 \tag{2.3}$$

Figure 2.1 (a) shows the real part of a simulated stepped-frequency waveform that goes through three frequency steps: 200Hz, 400Hz, and -100Hz. The results after computing the DFT of this signal are shown in Figure 2.1 (b).



(a) (b)

Figure 2.1. Simulated time series (a) and DFT (b) of three-frequency waveform

The DFT reveals signal energy at the correct frequencies of 200Hz, 400Hz, and -100Hz. However, all information on *when* a certain frequency occurred within the signal is lost in this display. The signal of Figure 2.1 (a) was 200 Hz from 0 to 5 seconds, 400 Hz from 5 to 10 seconds, and -100 Hz from 10 to 15 seconds. Doppler radar signal processing seeks to estimate the frequency, strength, and time-location of multi-component received signals. Due to the non-stationary (time-varying statistics) nature of radar waveforms, the DFT alone is unsuitable for Doppler processing.

2.2 INTRODUCTION TO JOINT TIME-FREQUENCY PROCESSING

Joint time-frequency (JTF) techniques have recently received considerable attention in the signal processing literature due to their ability to process non-stationary signals [26], [27]. Most time-frequency transforms fall into either the linear or quadratic classification. Some well known linear time-frequency transforms are the short-time Fourier transform (STFT) and the continuous wavelet transform (CWT). The Wigner-

Ville distribution (WVD) and the time-frequency distribution series (TFDS) are two common examples of quadratic time-frequency transforms.

In this work, the short-time Fourier transform (STFT) and its squared magnitude (spectrogram) are used for joint time-frequency processing. The spectrogram was chosen despite its two primary drawbacks. The first drawback is the time-frequency resolution limit imposed by the STFT window function (as discussed in the next section). Another disadvantage of the spectrogram is that it does not fully satisfy the mathematical properties of a time-frequency distribution. However, the spectrogram also has many advantages such as fast execution and intuitive interpretation that justify its use.

2.3 THE SHORT-TIME FOURIER TRANSFORM

The discrete short-time Fourier transform (STFT) of a signal $x[n]$ is defined as

$$STFT_x[k,n] = \sum_{r=-\infty}^{\infty} x[r]w[n-r]e^{-j2\pi rk/N}, \quad k = 0,1,\cdots,N-1 \tag{2.4}$$

where n is the discrete time index, k is the discrete frequency index, and $w[n]$ is a window function. The STFT can be viewed as the Fourier transform of a signal multiplied by a window function that slides in time. The duration of the window is typically chosen such that the signal of interest is approximately stationary over the duration of the window. A shorter duration window provides better time resolution at the cost of reduced frequency resolution. Similarly, a longer duration window offers better frequency resolution at the cost of degraded time resolution. This is the well-known time-frequency resolution tradeoff inherent in the STFT.

The spectrogram is used to extract useful signal parameters such as instantaneous frequency and bandwidth. The spectrogram of $x[n]$ is defined as the magnitude-squared of the STFT

$$Spectrogram_x[k,n] \triangleq \left|STFT_x[k,n]\right|^2 \tag{2.5}$$

Two important features of the spectrogram are that its values are (by definition) both real and nonnegative. To be classified as a time-frequency *distribution*, a function of time and frequency must be everywhere nonnegative and satisfy the corresponding marginals [26]. The time and frequency marginals are, respectively

$$\sum_k TFD_x[k,n] = \left|x[n]\right|^2 \tag{2.6}$$

$$\sum_n TFD_x[k,n] = \left|X[k]\right|^2 \tag{2.7}$$

where $TFD_x[k,n]$ corresponds to a generic time-frequency distribution. One of the primary disadvantages of the spectrogram is that it does *not* satisfy the time and frequency marginals. Despite the fact that the spectrogram is not a true time-frequency distribution, it is still quite useful in radar signal processing.

Figure 2.2 shows a spectrogram image of the stepped-frequency waveform in Figure 2.1 (a).

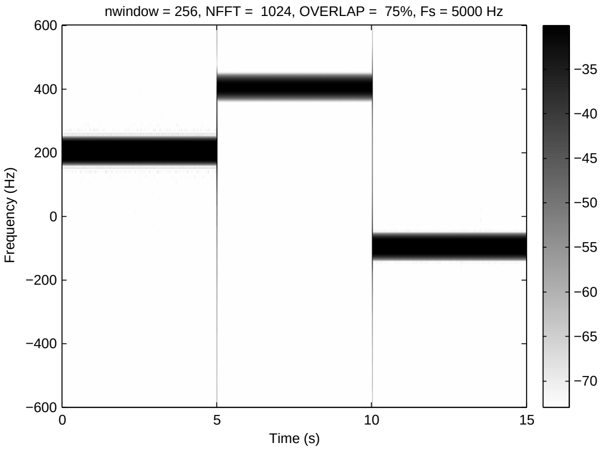


Figure 2.2. Spectrogram of stepped-frequency waveform

The sample rate of the original time-domain waveform was 5 kHz, and the spectrogram window duration was 256 points. The type of window used was the Kaiser window with beta equal to 7 and the size of each FFT was 1024 points. These are typical values used for the spectrograms in this dissertation. Along the horizontal axis is time in seconds while frequency in Hertz is along the vertical axis. The grayscale color shows the power (in dBm) of each signal component. As expected, the spectrogram reveals a 200 Hz signal from 0 seconds to 5 seconds, followed by a 400 Hz signal from 5 seconds to 10 seconds, and finally a -100 Hz component.

Chapter 3: The Micro-Doppler Phenomenon

3.1 INSTANTANEOUS FREQUENCY

Consider the scenario of Figure 3.1. A transmitted radar pulse $s_{TX}(t)$ of frequency f_0 travels a distance $R(t)$ to a target. The target scatters the pulse and the receiver processes a signal $s_{RX}(t)$ that is proportional to the time-delayed transmit pulse.

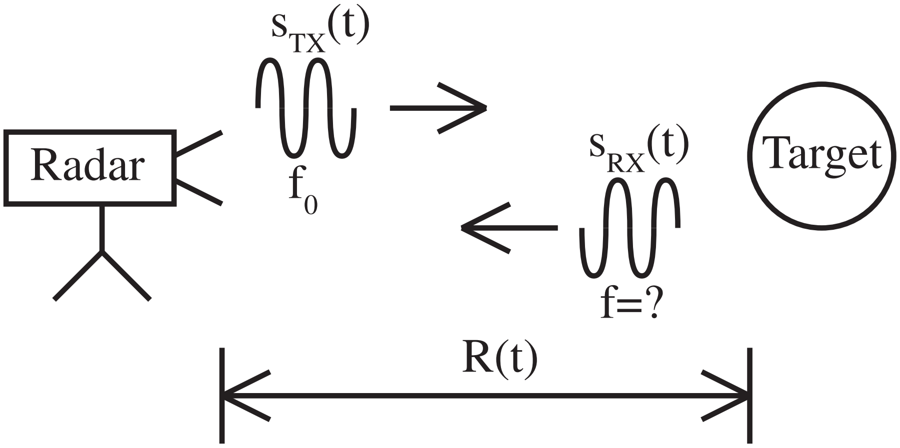


Figure 3.1. Radar configuration

For a single target moving relative to the radar, the round-trip time delay $\tau(t)$ becomes

$$\tau(t) = \frac{\text{distance traveled}}{\text{speed}} = \frac{2R(t)}{c} \tag{3.1}$$

where c is the speed of electromagnetic waves in the transmission medium and

$$R(t) = R_0 - v_R t - \frac{1}{2} a_R t^2 - \frac{1}{3} j_R t^3 + \textit{higher order terms} \tag{3.2}$$

Here R_0, v_R, a_R, and j_R are the position, velocity, acceleration, and jerk of the target relative to the radar (at the time of demodulation).

The signal present at the receiver input is

$$\begin{aligned} s_{RX}(t) &= \Gamma(t)\exp\{j[2\pi f_0(t-\tau(t))+\psi_0]\} \\ &= \Gamma(t)\exp\{j[2\pi f_0 t - \frac{4\pi f_0 R(t)}{c}+\psi_0]\} \\ &= \Gamma(t)\exp\{j\Phi(t)\} \end{aligned} \tag{3.3}$$

where ψ_0 is a constant phase offset at time t, $\Gamma(t)$ is the received signal amplitude, and $\Phi(t)$ is the phase. The instantaneous frequency of the received signal is proportional to the time derivative of the phase

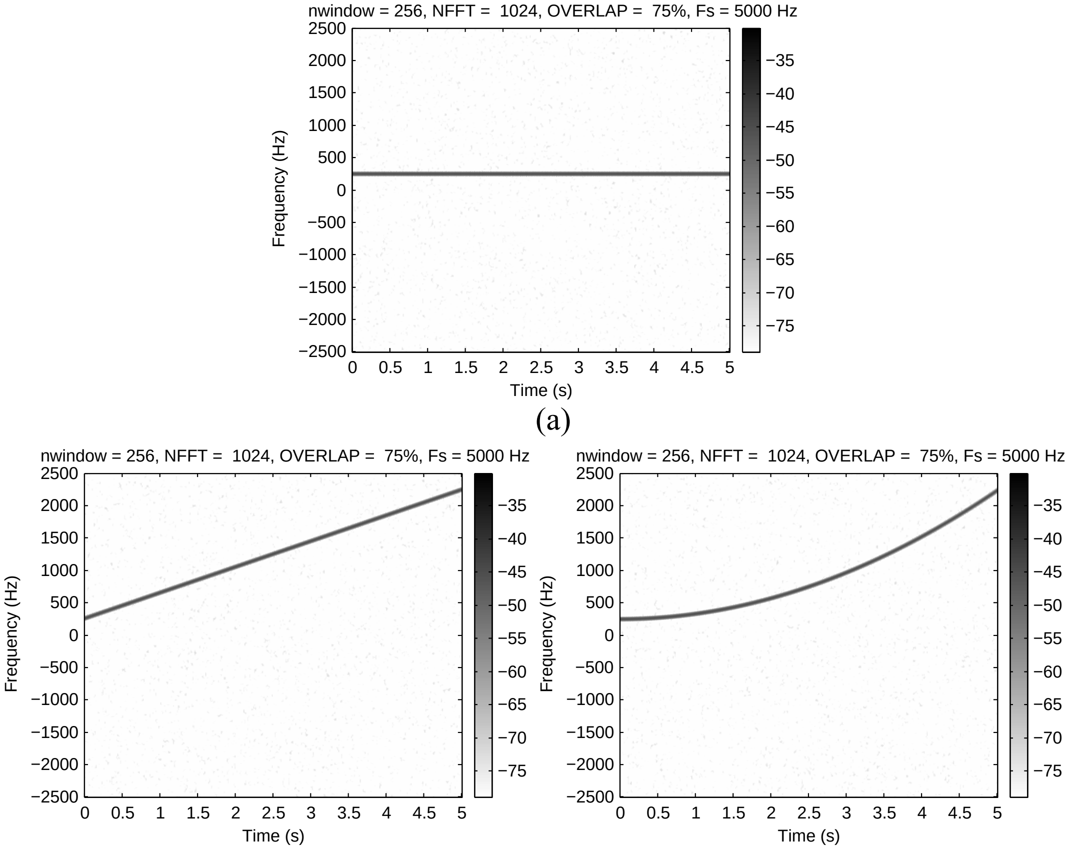
$$f_{inst}(t) = \frac{1}{2\pi}\frac{d\Phi(t)}{dt} = f_0 - \frac{2f_0}{c}\frac{dR(t)}{dt} = f_0 + \frac{2f_0 v_R}{c} + \frac{2f_0 a_R t}{c} + \frac{2f_0 j_R t^2}{c} \tag{3.4}$$

where cubic terms and higher have been neglected. The first term is the radar transmit frequency and the second term is the classical Doppler frequency. The third term results in linear frequency variation over time, while the fourth term results in quadratic frequency variation over time.

In a fully coherent radar system, the Doppler frequency is extracted by mixing the received signal with both an in-phase and a quadrature local oscillator of frequency f_0. Thus, the sign of the Doppler frequency is preserved, which distinguishes approaching from receding targets. The instantaneous frequency of the baseband signal is then:

$$f_{bb}(t) = \frac{2f_0 v_R}{c} + \frac{2f_0 a_R t}{c} + \frac{2f_0 j_R t^2}{c} \tag{3.5}$$

Figure 3.2 illustrates the effect of various simulated target motions. The SNR in each spectrogram is 15dB and the noise floor is -74 dBm.



(a)

(b)

(c)

Figure 3.2. Constant velocity (a), constant acceleration (b), and constant jerk (c)

In Figure 3.2 (a), a simulated target approaches the radar at 3.7 m/s (8.3 mph). This constant velocity results in a constant instantaneous frequency of 250 Hz. Figure 3.2 (b) shows a target approaching the radar with a constant acceleration of 6.0 m/s^2. Hence, quadratic phase terms result in linear instantaneous frequency terms. In Figure 3.2 (c), a target approaches the radar with a constant jerk of 1.2 m/s^3. As expected, cubic phase terms produce quadratic instantaneous frequency terms. These results will be crucial in interpreting the spectrograms of complex targets.

3.2 POINT SCATTERER MODEL AND EXAMPLE SPECTROGRAM

In order to gain intuition on the spectrograms of advanced targets, the point scatterer model is used. The point scatterer model is a widely used electromagnetic signal model. In essence, the point scatterer model takes a target with a continuum of scattering sites and reduces them to a small number of point scattering centers. Figure 3.3 illustrates a human walking toward a radar.

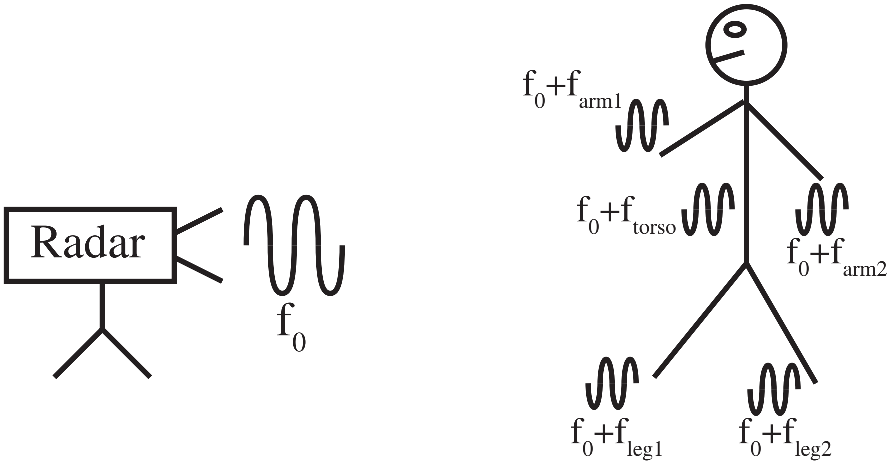


Figure 3.3. Human point scatterer model with five scattering centers

One possible point scatterer model would consolidate the scattering from a human target into five scattering centers. One scattering center would correspond to the returns from the torso, and each arm and leg would have its own scattering center. Every point scatterer in this model moves with its own velocity, and thus we associate individual Doppler frequency returns for each point scatterer. All Doppler returns not originating from the torso (non-rigid-body motion) are defined as *micro-Doppler*. Adding more scattering centers would result in a more accurate target representation [6], but the five scattering center model for a human is an acceptable compromise of accuracy, complexity, and intuition.

Figure 3.4 displays experimental radar data of a human running toward the radar.

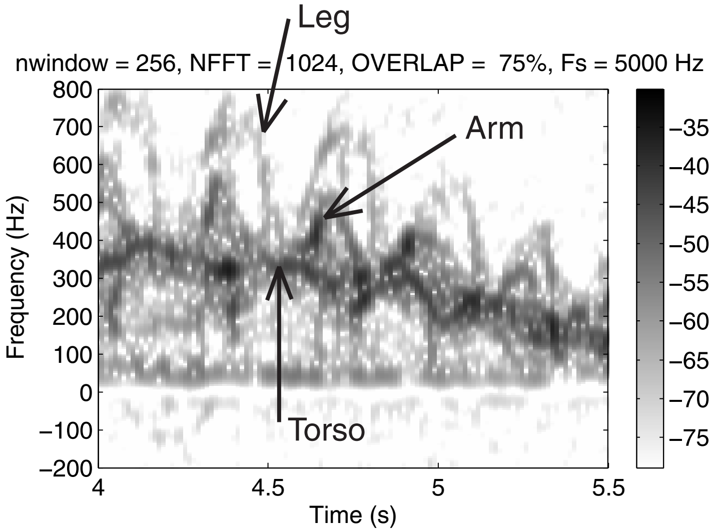


Figure 3.4. Micro-Doppler features of a running human

The strong, approximately linear return near the center of the spectrogram is due to the torso. The large looping returns come from leg motion. The smaller returns closer to the torso are from the arms. We see in Figure 3.4 that actual human radar data can be described quite well with a five scattering center approximation. This will be discussed further in Chapter 8.

Chapter 4: Sensor Selection and the MFCW Radar

4.1 SENSOR REQUIREMENTS

To support the primary objective, a goal for this dissertation was to design a relatively low-cost sensor to support detection and classification algorithm design. Two major application areas for this research are personnel detection and basic research. The sensor system requirements for this dissertation address the needs of both basic research and personnel detection applications. Table 4.1 lists the sensor system requirements.

Table 4.1. Sensor system requirements

Requirement Name	Value
90% detection range for all TOI	65 m (213 ft)
Average sensor false alarm rate	7 days
Maximum unambiguous range	125 m (410 ft), minimum
Azimuth angular coverage	20º, minimum
Per-unit production parts cost	less than $1000

The first requirement is the detection range for all targets of interest (TOI) given a 90% probability of detection and a fixed probability of false alarm. This probability of false alarm is set by the required average system false alarm rate. The specified minimum azimuth angular coverage is 20º. The target system per-unit parts cost is under $1,000.

4.2 CHOICE OF SENSOR

A multitude of sensor options exist for the moving target recognition (MTR) problem. This section will analyze competing sensor technologies. The list of sensors is by no means exhaustive, but it provides a cross-section of available technologies.

Buried cable sensors [28] exhibit superior performance in terms of detection and false alarm probabilities. A pair of “leaky” coaxial cables are buried underground and connected to a control unit. The first cable creates an electromagnetic field and the second cable senses disturbances in the field caused by objects moving through the detection region. Although buried cable sensors show excellent performance [29], installation is costly and time-consuming (and the system is essentially non-portable). Taut wire fence sensors [30] share similar performance to buried cable sensors, but they also share similar drawbacks.

A small number of land-based acoustic target classifiers [31], [32] have been developed. Land-based acoustic systems can be made low-cost and portable. Acoustic systems offer an alternative to Doppler radars. However, land-based acoustic systems suffer from limited range due to significant atmospheric attenuation of acoustic signals (1.3 dB per meter at 40kHz and 20dB per meter at 340kHz [31]).

Moving target recognition via optical video imaging has received considerable attention in recent years [33], [34], [35], and [36]. Optical video imaging systems classify targets by analyzing the optical flow of a video sequence. Under controlled conditions (minimal target occlusion, restricted angles of target motion, etc.), optical imaging systems can provide excellent recognition performance [34]. However, optical imaging system performance suffers under variation in lighting conditions, partial target occlusion, and varying range to the target. Optical imaging systems are typically costly due to the complex processing algorithms and expensive imagers required by many systems.

Passive infrared (PIR) sensors [37], [38] and passive millimeter wave (PMMW) sensors [38], [39] are two covert technologies for target detection. Both PIR and PMMW sensors detect the presence of objects such as humans, vehicles, and animals by their thermally emitted radiation. Unlike cabled sensors, PIR and PMMW sensor systems are often portable and capable of rapid deployment. Good detection performance can be obtained by both PIR and PMMW sensors in an interior environment. PMMW sensors also exhibit high detection performance in *exterior* environments. However, PIR sensor human detection performance degrades considerably in an exterior environment when ambient temperatures approach the temperature of the human body [38]. This phenomenon can be explained by the fact that the temperature contrast seen by a PIR sensor in the exterior environment is much lower than that of a PMMW sensor. The radiometric background temperature of the sky is around 300K at infrared frequencies and only approximately 20K at millimeter wave frequencies (27.7 GHz). Since the average human body temperature is 310K (98.6° F), the PIR sensor exhibits a lower temperature contrast than a PIR sensor (assuming the emissivities are not largely different). When the apparent temperature contrast between the background and target is small relative to the receiver's minimum detectable temperature shift, the receiver cannot reliably detect the target. Cryogenically cooling the sensor improves the receiver's minimum detectable temperature shift (sensitivity), but cooling cannot improve the temperature contrast between the target and background. Also, sensor cooling is not an attractive option since it adds significant cost and bulk to the system.

A common method employed to evade detection by passive (thermal) sensors is to wear a heavy coat. Although PIR sensor detection performance is significantly decreased by heavy clothing, PMMW sensors can actually perform slightly better when a light shirt is worn [38]. Millimeter wave frequencies penetrate clothing (and other non-metallic objects) much more than infrared. One drawback of both PMMW and PIR sensors is that an elevated false alarm rate can occur due to the reflection of thermal radiation by metallic building structures. In addition, some "thermal clutter" is produced by concrete, shrubs, and certain metallic building paints.

High range resolution (HRR) radars [40], [41] classify targets based on shape. Linear frequency modulation and stepped-frequency are the most common waveforms used for HRR radar. HRR radars provide superior target recognition performance for long-range classification tasks. HRR radars are capable of detecting, tracking, and classifying a multitude of targets. However, the high cost, power consumption, and computational demands of HRR radar often make them impractical for certain applications.

High Doppler resolution (HDR) radars [9], [42], [22] classify targets based on the dynamics of non-rigid-body motion. Some examples are the continuous wave (CW) and multiple frequency continuous wave (MFCW) radars. These high Doppler resolution systems utilize the micro-Doppler response for classifier development. Tracking multiple targets with HDR radars is more difficult than for HRR radars, but separation is possible when targets fall in separate Doppler bins. However, HDR radars can be many orders of magnitude less expensive than HRR radars. Unmatched Doppler resolution, low cost, and low power consumption make HDR radars attractive for short to medium range applications. The MFCW radar can utilize the micro-Doppler response and extract target range estimates with relatively low cost and computational burden. As a good tradeoff between cost, complexity, and performance, the MFCW radar was chosen for this dissertation research. The MFCW radar is the subject of the next section.

4.3 MULTIPLE FREQUENCY CONTINUOUS WAVE RADAR ARCHITECTURE

Figure 4.1 shows a simplified block diagram of one possible MFCW radar architecture.

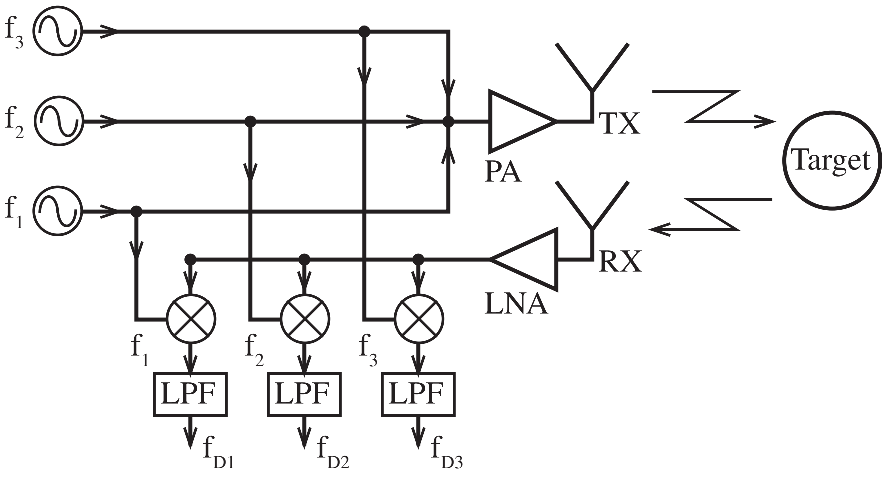


Figure 4.1. MFCW radar block diagram

The MFCW radar architecture of Figure 4.1 will be referred to as a three frequency continuous wave (3FCW) radar. As the name suggests, the 3FCW radar simultaneously transmits three different frequency tones. Three microwave frequency sources of frequency f_1, f_2, and f_3 provide the required tones. Although not shown in Figure 4.1, the three frequency sources are all phase-locked to the same crystal reference oscillator. Phase-locked oscillators provide excellent phase noise along with precise frequency spacing between the oscillators. Since all of the oscillators are locked to the same reference source, the frequencies tend to drift together which maintains the required frequency separation. The three tones are joined together in a Wilkinson power combiner (shown as a junction in Figure 4.1) and then passed through a power amplifier before transmission by the antenna. Since the radar is continuous wave (CW), one transmit and one receive antenna is used to provide enhanced isolation between the transmitter and receiver.

The 3FCW receiver is a direct-conversion receiver architecture. Thus, the incoming received signal is demodulated directly to baseband without an intermediate frequency. For illustration purposes, assume a received signal of the form

$$s_{RX}(t) = \cos(2\pi(f_1 + f_{D1})t) + \cos(2\pi(f_2 + f_{D2})t) + \cos(2\pi(f_3 + f_{D3})t) \tag{4.1}$$

where f_{D1}, f_{D2}, and f_{D3} are the Doppler frequencies modulated onto tone one, two, and three respectively. After passing through a low noise amplifier, the received signal power is split equally among the mixers. The signals at the mixers are then demodulated by local oscillators (LO) of frequency f_1, f_2, and f_3. Neglecting the amplitude factors, the output of the mixer that demodulates with LO frequency f_1 is

$$\begin{aligned}
\cos(2\pi f_1 t)s_{RX}(t) &= \cos(2\pi f_1 t)\cos(2\pi(f_1 + f_{D1})t) + \cos(2\pi f_1 t)\cos(2\pi(f_2 + f_{D2})t) \\
&\quad + \cos(2\pi f_1 t)\cos(2\pi(f_3 + f_{D3})t) \\
&= \frac{1}{2}\cos(2\pi f_{D1} t) + \frac{1}{2}\cos(2\pi(2f_1 + f_{D1})t) \\
&\quad + \frac{1}{2}\cos(2\pi(f_2 - f_1 + f_{D2})t) + \frac{1}{2}\cos(2\pi(f_2 + f_1 + f_{D2})t) \\
&\quad + \frac{1}{2}\cos(2\pi(f_3 - f_1 + f_{D3})t) + \frac{1}{2}\cos(2\pi(f_3 + f_1 + f_{D3})t)
\end{aligned} \tag{4.2}$$

The mixer outputs six different frequencies along with possibly some other unwanted spurious signals. Low-pass filtering the mixer output isolates the $\cos(2\pi f_{D1} t)$ term. The maximum expected Doppler frequency controls the choice of filter cutoff frequency. The assumption is made that $f_{D1}, f_{D2}, f_{D3} \ll (f_2 - f_1), (f_3 - f_1) \ll f_1, f_2, f_3$ so that the sum and difference frequency terms are rejected while the $\cos(2\pi f_{D1} t)$ term is retained. A similar derivation reveals that the output of the other two mixer/low-pass filter combinations is

$\cos(2\pi f_{D2} t)$ and $\cos(2\pi f_{D3} t)$. The receiver outputs three separate Doppler frequency channels.

Although neglected for clarity in Figure 4.1 and in the previous derivation, the three output channels are actually complex. Each of the three complex output signals contain an inphase (I) and quadrature (Q) component. The resulting complex baseband signal from the first mixer is $y_{bb1}(t) = I_1 + jQ_1$. Figure 4.2 shows how the I and Q channels are created by what is known as a quadrature demodulator.

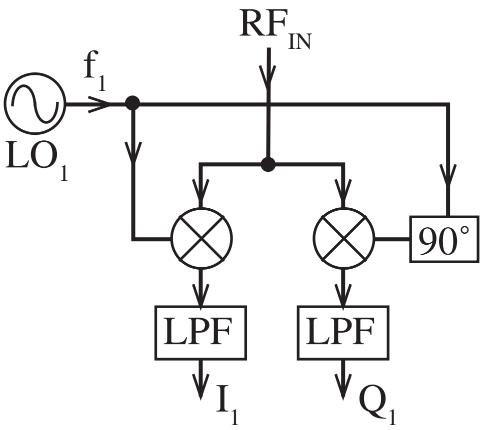


Figure 4.2. Quadrature demodulator

Figure 4.2 reveals that ideally the I and Q channels have the same frequency and amplitude, but are 90° out of phase. I/Q error occurs when the I and Q channels are not of equal amplitude or are not exactly 90° out of phase. Correction of I/Q error can be performed either in hardware or software. Software correction will be discussed in Chapter 7. The motivation behind quadrature demodulation is that it preserves the sign of the Doppler frequency. The Doppler frequency shift is positive for targets moving toward the radar and negative for targets moving away from the radar.

4.4 MFCW RADAR TARGET RANGE CALCULATION

As a tradeoff for the best possible Doppler resolution, standard continuous wave (CW) radars offer no useful range information. The CW radar's inability to offer useful range information stems from the unambiguous range problem. Reconsider equation (3.3) which gives the phase of a scattered target pulse at the receiver

$$s_{RX}(t) = \Gamma(t)\exp\{j[2\pi f_0 t - \frac{4\pi f_0 R(t)}{c} + \psi_0]\} \tag{4.3}$$

In principle, the range could be extracted from the return signal by phase comparing the transmit and receive signals [3]. The resulting signal phase is

$$\phi(t) = \frac{4\pi f_0 R(t)}{c} + \psi_0 . \tag{4.4}$$

After removal of the constant phase offset ψ_0 (by calibration), the range becomes

$$R(t) = \frac{c\phi(t)}{4\pi f_0} . \tag{4.5}$$

Since the phase is modulo 2π, the range becomes ambiguous when $\phi(t) = 2\pi m$, where m is a positive integer. Thus, the range becomes:

$$R = \frac{c2\pi m}{4\pi f_0} = \frac{cm}{2 f_0} \qquad m = 1,2,3,\cdots . \tag{4.6}$$

The maximum *unambiguous* range, R_{unamb}, occurs when m equals 1:

$$R_{unamb} = \frac{c}{2 f_0} . \tag{4.7}$$

For single-frequency CW radars operating at 10 GHz, R_{unamb} is 1.5cm. This radar cannot distinguish, for example, whether a target is at 0.5cm, 2cm, 3.5cm, or 150.5cm. For most applications, a maximum unambiguous range of 1.5cm is unacceptable. The multiple frequency continuous wave (MFCW) radar reduces the unambiguous range problem by simultaneously transmitting multiple tones of slightly different frequencies.

Now target range will be derived for an MFCW radar. For the MFCW radar where M is the number of tones, the received signal will be

$$s_{RX}(t) = \sum_{i=1}^{M} \Gamma_i(t) \exp\{j[2\pi f_i t - \frac{4\pi f_i R(t)}{c} + \psi_i]\} \tag{4.8}$$

where f_i, $\Gamma_i(t)$, and ψ_i are the frequency, amplitude factor, and constant phase offset of the ith tone, respectively. After complex demodulation, the phase $\phi_i(t)$ of the ith tone is

$$\phi_i(t) = \frac{4\pi f_i R(t)}{c} - \psi_i \,. \tag{4.9}$$

The phase difference between tones p and q is then

$$\Delta\phi_{p,q}(t) = \frac{4\pi \Delta f_{p,q} R(t)}{c} - \Delta\psi_{p,q} \tag{4.10}$$

where $\Delta f_{p,q} = \left| f_p - f_q \right|$ is the absolute frequency difference and $\Delta\phi_{p,q}(t) = \phi_p(t) - \phi_q(t)$ is the phase difference modulo 2π between tones p and q. The constant phase offset between tones p and q is $\Delta\psi_{p,q} = \psi_p - \psi_q$. Removal of the phase offset from $\Delta\phi_{p,q}(t)$ by calibration is necessary to ensure proper range results. One calibrates the radar by comparing the estimated target range with the true range, thus calculating $\Delta\psi_{p,q}$ in equation (4.10). After initial calibration, recalibration is typically not required unless the hardware configuration changes (changes in relative phase among the signal paths force a recalibration). Following calibration, the range calculated from tones p and q is

$$R_{p,q}(t) = \frac{c\Delta\phi'_{p,q}(t)}{4\pi \Delta f_{p,q}} \tag{4.11}$$

where $\Delta\phi'_{p,q}(t)$ is the phase difference after calibration. The range $R_{p,q}(t)$ becomes ambiguous when $\Delta\phi'_{p,q}(t) = 2\pi m$ where m is a positive integer. Hence the maximum unambiguous range due to tones p and q is

$$R_{unamb;p,q} = \frac{c}{2\Delta f_{p,q}} \tag{4.12}$$

The theoretical root mean square (rms) range error due to tones p and q is [43]

$$\delta R_{p,q} = \frac{R_{unamb;p,q}}{2\pi(2SNR)^{1/2}} \tag{4.13}$$

where SNR is the signal to noise ratio.

Equations (4.12) and (4.13) show that both the maximum unambiguous range and range accuracy are inversely proportional to the frequency difference. A tradeoff between maximum unambiguous range and range accuracy occurs if only two transmit frequencies are used. Longer maximum unambiguous range requirements impose coarse range accuracy. This motivates the use of three or more frequencies in MFCW radars. An MFCW radar with more than two tones offers improved unambiguous range and range accuracy. The MFCW radar unambiguous range becomes the maximum of the unambiguous ranges between any two frequency pairs

$$R_{unamb;MFCW} = \max_{p,q} R_{unamb;p,q} \tag{4.14}$$

After resolving ambiguities [3], the theoretical MFCW root mean square (RMS) range error is the minimum error of any two frequency pairs

$$\delta R_{MFCW} = \min_{p,q} \delta R_{p,q} \tag{4.15}$$

For MFCW radars with three or more tones, the tradeoff for performance enhancement is added system complexity and cost.

For clarification, the range parameters will be computed for the radar design used in this dissertation (three transmit frequencies). Let tone one with frequency f_1 and tone two with frequency f_2 be two tones closely spaced in frequency. Tone three with frequency f_3 is spaced farther apart from tone one than tone two as illustrated in Figure 4.3.

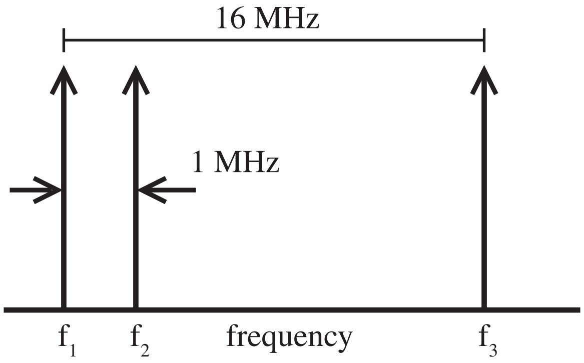


Figure 4.3. Three tone frequency separation

Tones one and two provide a long unambiguous range while tones one and three offer an accurate but ambiguous range. Table 4.2 lists the parameters for this example.

Table 4.2. Ranging parameters

Parameter	Value
$\Delta f_{1,2}$	1 MHz
$\Delta f_{1,3}$	16 MHz
f_1	10 GHz
SNR	13.5 dB
$R_{unamb;1,2}$	150 m (492 ft)
$R_{unamb;1,3}$	9.4 m (30.8 ft)
$\delta R_{1,2}$	3.6 m (11.7 ft)
$\delta R_{1,3}$	0.22 m (0.73 ft)

The tones separated by 1MHz provide a maximum unambiguous range of 150m with a 3.6m RMS range error (theoretical). The 16MHz separated tones provide a maximum unambiguous range of 9.4m with a 0.22m RMS range error. The range accuracy calculations used 13.5dB as the SNR. The combined use of all three tones offers a maximum unambiguous range of 150m with a 0.22m RMS range error.

Chapter 5: Target Detection and Ranging

5.1 RADAR RANGE EQUATION

The calculation of fundamental radar design parameters from system requirements typically starts with the use of the radar range equation. One of many forms of the radar range equation is

$$R_{\max} = \left[\frac{G_t G_r P_t \lambda^2 \sigma}{(4\pi)^3 k T_0 B F SNR_{\min} L} \right]^{1/4} \tag{5.1}$$

where G_t = transmitting antenna gain

G_r = receiving antenna gain

P_t = transmit power

λ = wavelength of transmit signal

σ = target radar cross section

k = Boltzmann's constant (1.38×10^{-23} J/K)

T_0 = room temperature (290K)

B = receiver bandwidth

F = receiver noise figure

L = system loss factor

$SNR_{\min}$ = minimum signal to noise ratio required for detection

Equation (5.1) calculates the maximum detectable range to a target of radar cross section (RCS) σ. The radar range equation is an over-simplification of a complex problem, but it gives the system designer a good starting point. One notable simplification is that the RCS value is a constant. A target's RCS varies with aspect angle, frequency, and polarization. This form of the radar range equation also neglects received signal fluctuation due to dynamic propagation effects such as multipath.

Equation (5.1) in its current form gives no direct information as to the reliability of detection. The parameters that describe detection reliability (probability of detection and false alarm) are contained in the $SNR_{\min}$ term. In the next section, the important concepts of classical detection theory will be reviewed and utilized to explicitly express $SNR_{\min}$ in terms of probability of detection and false alarm.

5.2 CLASSICAL DETECTION THEORY

Signal detection is a classical problem of binary statistical hypothesis testing. Under the null hypothesis H_0, the complex baseband received signal $y(t)$ is composed of noise alone.

$$H_0: \quad y(t) = n(t) \qquad \text{where} \quad n(t) = n_I(t) + jn_Q(t) \tag{5.2}$$

The predominant noise source is assumed to be thermal, so both the in-phase and quadrature noise components are Gaussian. Under hypothesis H_1, the received signal $y(t)$ is the sum of the transmitted signal $s(t)$ and noise.

$$H_1: \quad y(t) = s(t) + n(t) \quad \text{where} \quad s(t) = s_I(t) + js_Q(t) \tag{5.3}$$

Let I and Q be the in-phase and quadrature components of $y(t)$, respectively. For this development, the received signal is processed by a linear detector

$$V = \sqrt{I^2 + Q^2} \tag{5.4}$$

where V is the complex envelope of the baseband signal. I and Q are independent Gaussian random variables under hypothesis H_0. Under H_0, the complex envelope V is Rayleigh distributed

$$p_{V|H_0}\left(v \mid H_0\right) = \frac{v}{\sigma^2} \exp\left[\frac{-v^2}{2\sigma^2}\right] \tag{5.5}$$

where σ^2 is the conditional variance and $p_{V|H_0}(v \mid H_0)$ is the conditional probability density function of V given that the received signal is only noise (H_0). Equation (5.5) depends on the single parameter σ^2. Estimation of σ^2 is facilitated by the following relation

$$\sigma = \sqrt{\frac{2}{\pi}} E\left[V \mid H_0\right] \tag{5.6}$$

where $E\left[V \mid H_0\right]$ is the conditional expected value of V given H_0.

S.O. Rice showed that under hypothesis H_1, the envelope V takes on the Ricean distribution [44]

$$p_{V|H_1}\left(v \mid H_1\right) = \frac{v}{\sigma_{V|H_1}{}^2} \exp\left[\frac{-\left(v^2 + A^2\right)}{2\sigma_{V|H_1}{}^2}\right] I_0\left(\frac{vA}{\sigma_{V|H_1}{}^2}\right) \tag{5.7}$$

where $I_0(\bullet)$ is the modified Bessel function of zero order. Here $\sigma_{V|H_1}{}^2$ is the conditional variance and $p_{V|H_1}\left(v \mid H_1\right)$ is the conditional probability density function of V given that the received signal is a sinusoidal signal of amplitude A plus noise. Figure 5.1 shows the probability density functions under H_0 and H_1.

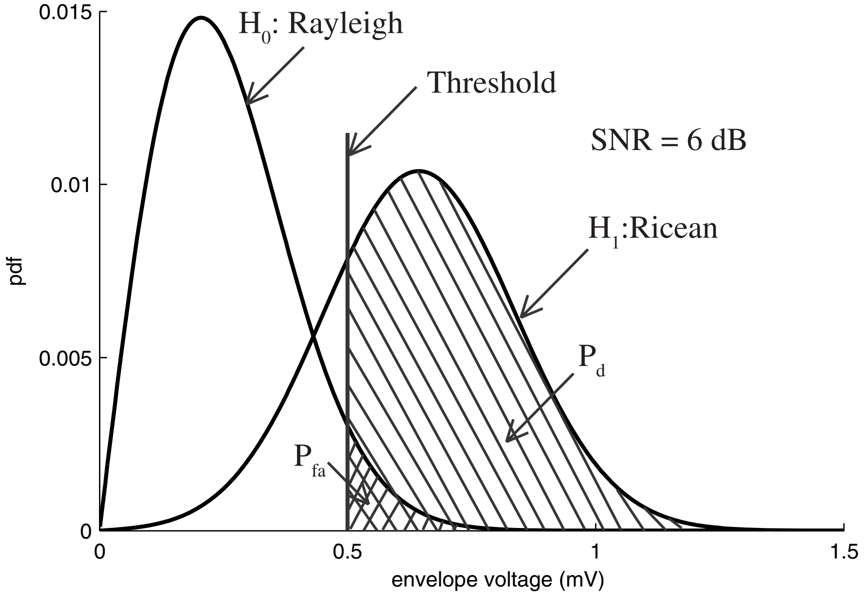


Figure 5.1. Probability density functions of noise and signal-plus-noise

Two types of errors can occur in the detection problem. A type 1 error, commonly known as a false alarm, happens when no signal is present but the noise level exceeds the detection threshold. A type 2 error, known as a missed detection, occurs when a signal is present but the signal level is not high enough to cross the detection threshold. Intuition indicates the desire to simultaneously minimize both types of errors. The total error is minimized in the ideal observer detection criterion [43] which sets the detection threshold at the intersection of the probability density functions. This detection methodology often works well in communications systems where the cost of making either type of error is typically equal. However, in radar systems, the ideal observer criterion results in high false alarm rates. The Neyman-Pearson detection criterion provides a method of detection that is well suited to the radar problem.

In the Neyman-Pearson detector, the probability of false alarm P_{fa} is fixed, while the probability of missed detection P_m is minimized. Equivalently, the probability of detection P_d is maximized since $P_d = 1 - P_m$. The probability of false alarm is typically fixed by system requirements on the minimum tolerable time between false alarms. For continuous-time detection, it can be shown [43] that the probability of false alarm is related to the average time between false alarms by

$$P_{fa} = \frac{1}{T_{fa} B} \tag{5.8}$$

where T_{fa} is the average time between false alarms and B is the receiver noise bandwidth (roughly approximated by the 3dB bandwidth of the pre-detection filters). Once the system designer determines the noise bandwidth, the probability of false alarm is fixed by the required average time between false alarms.

The detection threshold is easily obtained once the probability of false alarm is set. As seen in Figure 5.1, the probability of false alarm is the area under the noise-only curve above the threshold level γ.

$$P_{fa} = \int_{\gamma}^{\infty} p_{V|H_0}(v \mid H_0)\,dv = \int_{\gamma}^{\infty} \frac{v}{\sigma^2} \exp\left[\frac{-v^2}{2\sigma^2}\right] dv = \exp\left[\frac{-\gamma^2}{2\sigma^2}\right] \tag{5.9}$$

Solving for γ yields

$$\gamma = \sigma \sqrt{2 \ln\left(\frac{1}{P_{fa}}\right)} \tag{5.10}$$

If the noise statistics are not time-varying, the threshold γ maintains the specified probability of false alarm while maximizing the probability of detection. In order to maintain a constant probability of false alarm in the presence of non-stationary noise, an adaptive threshold method such as the constant false alarm rate (CFAR) detector is required.

As shown in Figure 5.1, the probability of detection is the area under the signal-plus-noise curve above the detection threshold.

$$P_d = \int_{\gamma}^{\infty} p_{V|H_1}(v \mid H_1)dv = \int_{\gamma}^{\infty} \frac{v}{\sigma^2} \exp\left[\frac{-(v^2 + A^2)}{2\sigma^2}\right] I_0\left(\frac{vA}{\sigma^2}\right)dv \tag{5.11}$$

The integral of equation (5.11) cannot be solved by traditional analytical techniques. The numerical solution of equation (5.11) is [45]

$$P_d = \frac{1}{2} erfc\left(\sqrt{-\ln\left(P_{fa}\right)} - \sqrt{SNR + \frac{1}{2}}\right) \tag{5.12}$$

where $erfc(\bullet)$ is the complimentary error function. Solving for SNR gives

$$SNR = \left(\sqrt{\ln\left(\frac{1}{P_{fa}}\right)} - erfc^{-1}\left(2P_d\right)\right)^2 - \frac{1}{2} \tag{5.13}$$

where $erfc^{-1}(\bullet)$ is the inverse complimentary error function. Figure 5.2 plots the receiver operating curve (ROC), a family of solutions to equation (5.12).

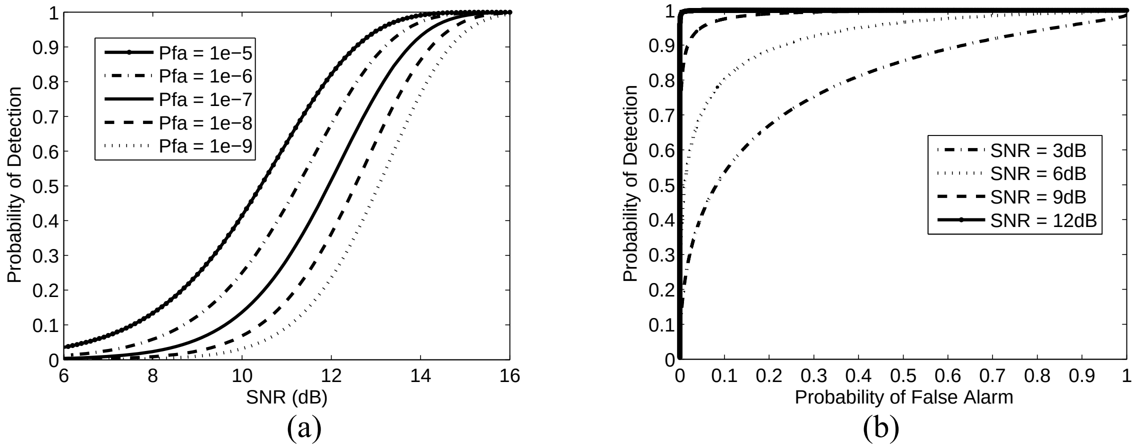


Figure 5.2. Probability of detection versus SNR (a), and receiver operating curve (b)

Figure 5.2 (a) provides a quick view of the impact of varying system requirements on the required SNR. Lowering the false alarm rate (increasing the average time between false alarms) results in higher required SNRs for the same probability of detection. Also, if the required probability of detection is reduced while maintaining the same false alarm rate, lower SNRs are required.

Equation (5.13) can now be substituted into equation (5.1) to get a radar range equation which explicitly includes detection reliability terms

$$R_{\max;P_d,P_{fa},\sigma} = \left[\frac{G_t G_r P_t \lambda^2 \sigma}{(4\pi)^3 k T_0 B F \left\{ \left(\sqrt{\ln\left(\frac{1}{P_{fa}}\right)} - erfc^{-1}\left(2P_d\right) \right)^2 - \frac{1}{2} \right\} L} \right]^{1/4} \qquad (5.14)$$

where $R_{\max;P_d,P_{fa},\sigma}$ is the maximum range to a target of radar cross section σ, given that the required probability of detection is P_d and the required probability of false alarm is P_{fa}. It is important to remember that equation (5.14) is still an approximation of the maximum range because of received signal and noise statistical fluctuations. The effects of fluctuations can be reduced by averaging (integrating) the received signal over a short time window.

5.3 DETECTION IN THE JOINT TIME-FREQUENCY DOMAIN

The detection method described in the previous section historically has been performed in either the time or frequency domains. Recently, joint time-frequency domain detection has received increasing attention [27], [46], [47]. The reason for the rising interest is attributed to the fact that uncorrelated noise spreads out over the time-frequency domain, while frequency-modulated signals tend to localize. With a suitably defined SNR [27], the joint time-frequency domain SNR will be greater than (or equal to in certain cases [47]) the SNR in either the time or frequency domains. Since Doppler radar signatures primarily consist of frequency-modulated components, performance is likely to improve by detecting in the joint time-frequency domain.

The detection theory developed in Section 5.2 is still valid in the time-frequency domain if we make some adjustments. Equation (5.4) for the received signal envelope should be changed to

$$V_{k,n} = \sqrt{(\mathrm{Re}[C_{k,n}])^2 + (\mathrm{Im}[C_{k,n}])^2} \tag{5.15}$$

where $\mathrm{Re}[C_{k,n}]$ and $\mathrm{Im}[C_{k,n}]$ are the real and imaginary parts of time-frequency coefficient $C_{k,n}$. In the time-frequency domain, the probability of false alarm P_{fa} and the probability of detection P_d are now specified per spectrogram "pixel." In other words, P_{fa} should be interpreted as the probability that a single time-frequency bin false alarms. To reduce the overall *system* false alarm probability $P_{fa,sys}$, a radar system would likely not report an alarm unless more than a single time-frequency bin was detected. The quantity P_{fa} will be referred to as the probability of *pixel* false alarm and $P_{fa,sys}$ will be referred to as the probability of *system* false alarm. Let us now derive the probability of system false alarm.

Each column of a spectrogram corresponds to an N - point fast Fourier transform (FFT) of a windowed segment of data. One method of reducing the overall system false alarm rate is to require a minimum number of detected pixels within a given FFT before signaling a system alarm. If at least z detected pixels are required to set a system alarm, the probability of system false alarm is the probability that z or more pixels cross the detection threshold when no target is present.

Let M be one of N FFT points following *pixel-level* detection. If no target is present, M takes on a Bernoulli distribution of parameter P_{fa}

$$f_M(m) = P_{fa}{}^m (1 - P_{fa})^m \quad \text{where } m \in \{0,1\} \text{ and } P_{fa} \in [0,1] \tag{5.16}$$

where $f_M(m)$ is the Bernoulli probability mass function of M. Hence, with probability P_{fa}, M is a false alarm, and with probability $(1 - P_{fa})$, M is not a false alarm. The probability that exactly r (independent) pixels false alarm in N trials is given by the binomial distribution for N fixed

$$P(R = r) = f_{R|N}(r \mid N) = \binom{N}{r} P_{fa}{}^r (1 - P_{fa})^{N-r} \tag{5.17}$$

where $P(\bullet)$ is a suitable probability measure and $\binom{N}{r}$ is a binomial coefficient. To determine the probability of system false alarm $P_{fa,sys}$, the probability that z or more pixels false alarm must be known. The resulting probability of system false alarm is

$$P_{fa,sys} = P(R \geq z \mid N) = \sum_{j=z}^{N} \binom{N}{j} P_{fa}{}^j (1 - P_{fa})^{N-j} = I_{P_{fa}}(z, N - z - 1) \tag{5.18}$$

where $I_x(a,b)$ is the regularized incomplete beta function [48].

In a system design, the designer calculates the probability of system false alarm from the specified minimum average time between false alarms. If T_{FFT} is the time in seconds between consecutive FFTs, the number of *system-level* detection decisions per second is $\frac{1}{T_{FFT}}$. The minimum average time between false alarms, $T_{fa,sys}$, is then

$$T_{fa,sys} = \frac{T_{FFT}}{P_{fa,sys}} \tag{5.19}$$

After calculating $P_{fa,sys}$ from $T_{fa,sys}$, the designer obtains P_{fa} from

$$P_{fa} = I^{-1}_{P_{fa,sys}}(z, N-z+1) = I^{-1}_{\frac{T_{FFT}}{T_{fs,sys}}}(z, N-z+1) \tag{5.20}$$

where $I^{-1}_{P_{fa,sys}}(z, N-z+1)$ is the inverse regularized incomplete beta function [49]. The probability of pixel false alarm is then used to find the minimum required signal to noise ratio. Table 5.1 lists the parameters for the radar system used in this dissertation.

Table 5.1. Detection parameters

Name	Symbol	Value
mean time between system false alarms	$T_{fa,sys}$	$7\ days$
time between each FFT	T_{FFT}	$38.5ms$
required number of pixels for alarm	z	2
probability of system false alarm	$P_{fa,sys}$	6.37×10^{-8}
probability of pixel false alarm	P_{fa}	3.49×10^{-7}
probability of detection	P_d	0.9
Minimum SNR required for detection	$SNR_{\min,P_d,P_{fa}}$	$13.5dB$

Starting with a specification of 7 days between system false alarms, the probability of pixel false alarm is calculated as $P_{fa} = 3.49\times10^{-7}$. For a requirement of 0.9 probability of detection, the minimum SNR for detection is 13.5dB (see Section 5.2).

Requiring z or more pixels for detection to reduce the false alarm rate does not go without penalty. Suppose that z equals 2 and P_d equals 0.9. If target signal energy appears in only one frequency bin, the radar has no chance of detecting the target. If at least two frequency bins contain target energy (at the minimum SNR), the probability of system-level detection increases to approximately 0.81 (assuming independent samples).

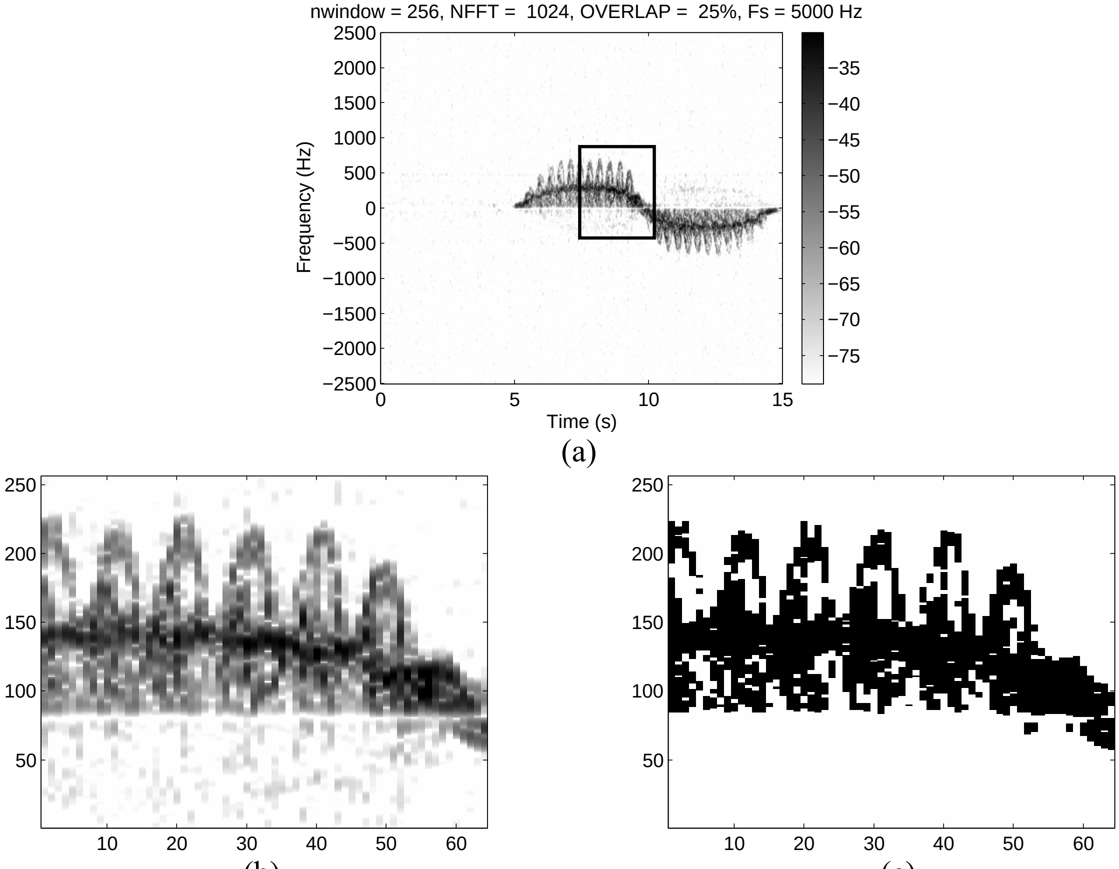
When there are at least three frequency bins, the system-level detection probability increases to approximately 0.97. In general, if Q independent frequency bins contain target energy at the minimum required SNR, the system-level probability of detection is

$$P_{d,sys} = \sum_{j=z}^{Q} \binom{Q}{j} P_d^{\,j} (1 - P_d)^{Q-j} = I_{P_d}(z, Q - z - 1) \tag{5.21}$$

Although probability of detection loss occurs for narrowband targets (targets whose frequency spread is on the order of the frequency resolution), the benefits of reduced probability of system false alarm outweigh this loss. The detection algorithm can exploit the inherent frequency spread due to micro-Doppler to improve detection performance.

5.4 PROCEDURE FOR JOINT TIME-FREQUENCY DOMAIN DETECTION

Figure 5.3 (a) shows the spectrogram of a human target. The target jogs toward the radar going from a range of 100ft to 50ft and then turns around and jogs back to 100ft.



(a)

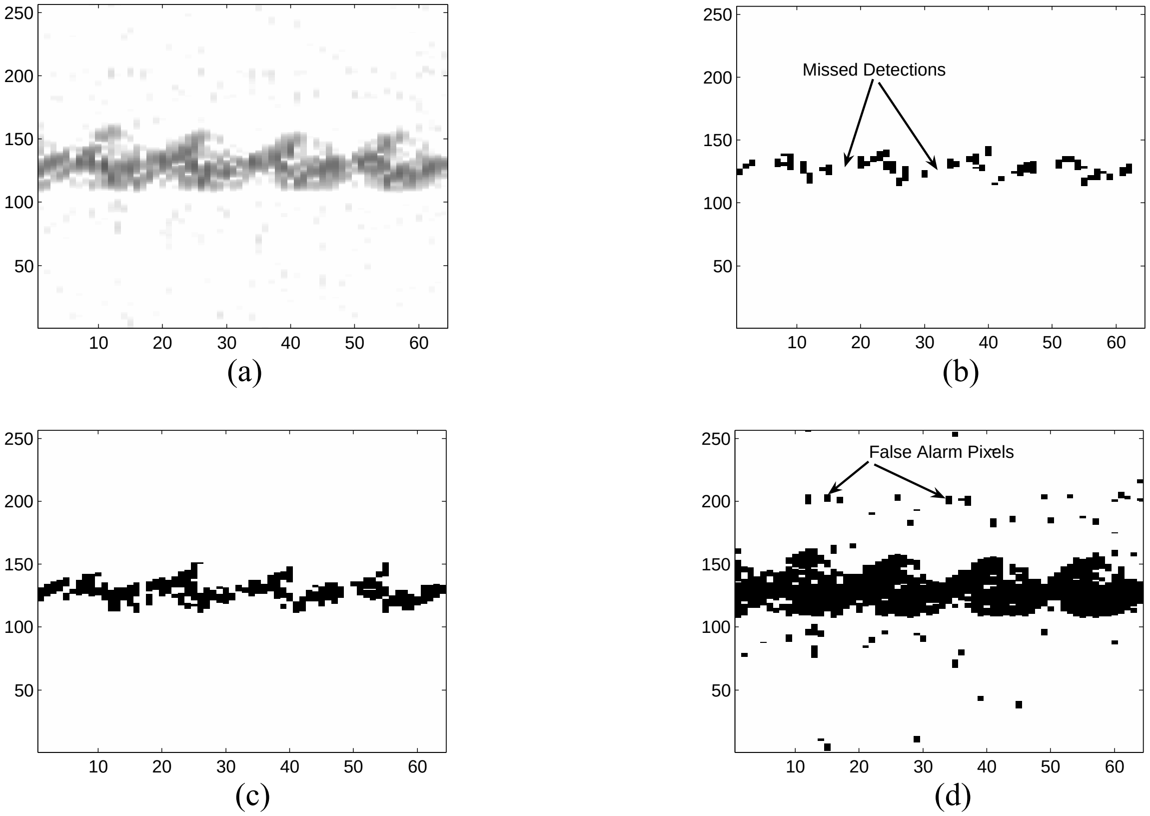
(b)

(c)

Figure 5.3. Complete spectrogram (a), extracted frame (b), and detected frame (c)

The detection procedure starts by extracting a frame (typically 1,250Hz by 2.46s) centered on the strongest spectrogram return within a given time frame (see Section 5.5 for target tracking). An extracted frame is shown in Figure 5.3 (b). Each pixel within the frame is compared with a threshold to determine if noise or target energy is present within that pixel. Figure 5.3 (c) displays the detected frame for a threshold of -61dBm. The detected frame shows good detection of the target torso along with the micro-Doppler produced by the legs and arms.

Figure 5.4 illustrates the effect of varying the detection threshold on pixel false alarms and missed detections.



(a) (b) (c) (d)

Figure 5.4. Spectrogram frame for human target (a), detection results for threshold level: -55dBm (b), -61dBm (c), and -76dBm (d)

Figure 5.4 (a) is a spectrogram frame of a human target walking toward the radar at a range of approximately 225ft. In Figure 5.4 (b) the detection threshold is set to -55dBm. No pixels false alarm within this detection frame, but numerous missed detections occur. In Figure 5.4 (d), a much higher threshold of -76dBm is used. This threshold dramatically increases the number of false alarm pixels but considerably decreases the number of missed detections. The threshold of -61dBm used in Figure 5.4 (c) achieves an acceptable balance between the number of false alarm pixels and missed detection pixels.

5.5 RANGING IN THE JOINT TIME-FREQUENCY DOMAIN: THE TIME-FREQUENCY-RANGE DIAGRAM (TFRGRAM)

In Section 4.4, the MFCW radar target range was calculated from the phase difference between any pair of tones. An analog phase comparator could be used to extract the phase difference between two Doppler channels. However, to resolve multiple targets by their Doppler frequencies, a phase comparator would need to be placed on each output of a Doppler filter bank (unless one time-shared filter is to be used). The expense and complexity of such a system grows rapidly as the requirement for finer Doppler resolution tightens. Another option is to calculate the phase difference digitally in software. One could perform a phase estimate in the time domain, but the ability to resolve multiple targets by Doppler would be lost. In this dissertation, the phase difference estimate is performed in the joint time-frequency domain.

The STFT magnitude (spectrogram) is already being used to perform detection and signature extraction, so the STFT phase provides the phase difference estimate with little extra computation. The STFT can be split into magnitude and phase components

$$STFT_x[k,n] = \left|STFT_x[k,n]\right| \angle STFT_x[k,n] \ . \tag{5.22}$$

If $\mathrm{Re}[C_{k,n}]$ and $\mathrm{Im}[C_{k,n}]$ are the real and imaginary parts of time-frequency coefficient $C_{k,n}$, then

$$\begin{aligned} \left|STFT_x[k,n]\right| &= \sqrt{(\mathrm{Re}[C_{k,n}])^2 + (\mathrm{Im}[C_{k,n}])^2} \\ \angle STFT_x[k,n] &= \tan^{-1}\left[\frac{\mathrm{Im}[C_{k,n}]}{\mathrm{Re}[C_{k,n}]}\right] \end{aligned} \tag{5.23}$$

where the inverse tangent must be a four quadrant operation for proper results. The phase difference between tones p and q at frequency index k and time index n is

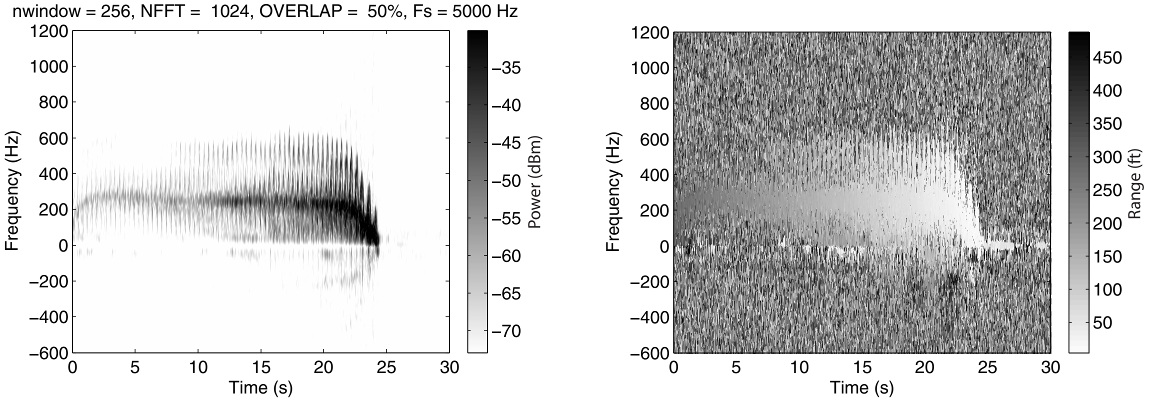
$$\Delta\phi_{p,q}[k,n] = \angle STFT_p[k,n] - \angle STFT_q[k,n] \quad \text{where } \Delta\phi_{p,q}[k,n] \in [0, 2\pi] \tag{5.24}$$

In order to ensure that $\Delta\phi_{p,q}[k,n] \in [0, 2\pi]$, the arithmetic for the phase calculation must be modulo 2π. The phase difference between tones p and q gives

$$R_{p,q}[k,n] = \frac{c\Delta\phi_{p,q}[k,n]}{4\pi\Delta f_{p,q}} \tag{5.25}$$

where $R_{p,q}[k,n]$ is the range estimate at frequency index k and time index n. To prevent adding bias to $R_{p,q}[k,n]$, it is important to remove any constant phase offset from $\Delta\phi_{p,q}[k,n]$ (by range calibration) prior to calculating $R_{p,q}[k,n]$.

The TFRgram (Time-Frequency-Range diagram) introduced by this dissertation is a novel method of visualizing and extracting MFCW radar range information. The TFRgram is the time-frequency-range image of $R_{p,q}[k,n]$. Figure 5.5 plots the spectrogram and TFRgram of a human target jogging from 292 feet to 15 feet.



(a) (b)

Figure 5.5. Spectrogram (a) and TFRgram (b) of human jogging from 292ft to 15ft

The TFRgram is similar to a spectrogram in that the target is located in the same time-frequency bins; however, the color axis plots range instead of power. As expected, the range appears random in regions of the time-frequency plane where no target is present. Valid estimates of target range only come from regions of $R_{p,q}[k,n]$ with target signal energy. A threshold mask (calculated from the spectrogram) can be applied to the TFRgram to extract only the range estimates from detected targets. Figure 5.6 shows a TFRgram after the application of a threshold mask (with inverted color mapping).

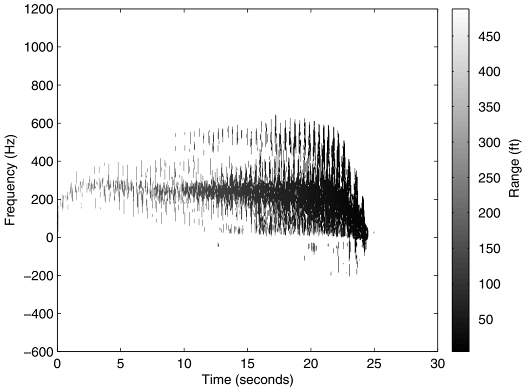


Figure 5.6. TFRgram after application of a threshold mask

In order to extract range estimates from the TFRgram, the frequency coordinates of the target must be found. The frequency coordinates select the proper region of the TFRgram to extract range information. Due to the Doppler effect, note that the frequency coordinates are proportional to the target velocity relative to the radar. One possible extraction method calculates the frequency location of the maximum value of the spectrogram at each time instant

$$k_{\text{target}}[n]\Big|_{n=n_0} = \underset{\text{k}}{\text{argmax}}\left[\left|STFT_x[k,n]\right|^2\Big|_{n=n_0}\right] \tag{5.26}$$

where $k_{\text{target}}[n]$ is the frequency coordinate of the target at time n. A scaled version of $k_{\text{target}}[n]$ is plotted in Figure 5.7 (a) for the human target of Figure 5.5.

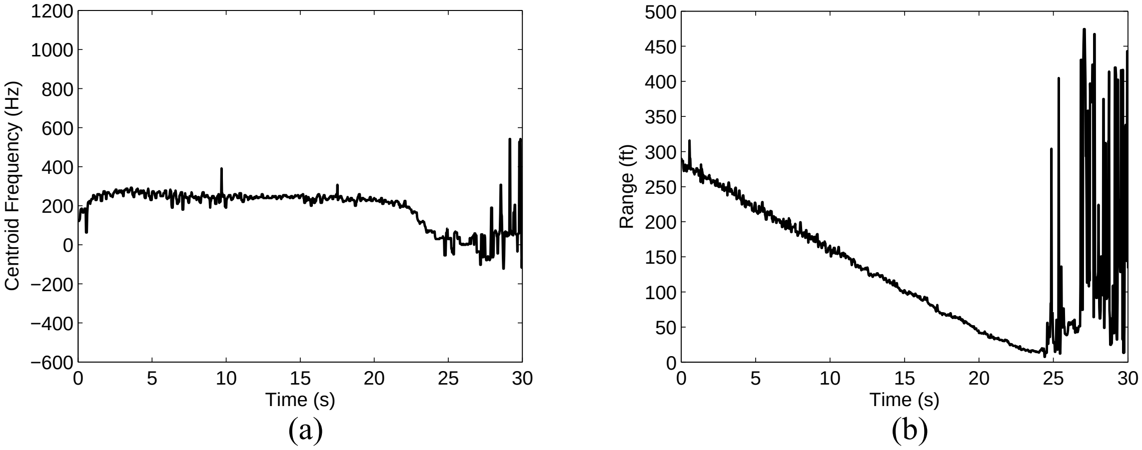


Figure 5.7. Centroid frequency (a) and target range (b)

The target frequency coordinate is tracked reasonably well until the target stops moving at about the 25 second mark. The frequency coordinate should be zero here, but instead, the value fluctuates rapidly due to signal loss. This discrepancy can be corrected by zeroing the frequency coordinate at any time instant that no signal is detected.

Figure 5.7 (b) shows the target range extracted from the TFRgram using the frequency coordinates of Figure 5.7 (a). Two tones separated in frequency by 1MHz were used for this calculation. Since rms range accuracy improves with increasing SNR, the range estimate fluctuates less as the target approaches the radar (until the target stops moving at around the 25 second mark).

Chapter 6: Radar Hardware Design

6.1 RADAR SYSTEM-LEVEL DESIGN

Most radar system designs begin with a list of requirements similar to those of Table 4.1. A radar designer then typically consults an appropriate range equation to determine which parameters are fixed by requirements and which parameters the designer can manipulate. The range equation used for this design will be equation (5.14) of Section 5.2 repeated below

$$R_{\max;P_d,P_{fa},\sigma} = \left[\frac{G_t G_r P_t \lambda^2 \sigma}{(4\pi)^3 k T_0 B F \left\{ \left(\sqrt{\ln\left(\frac{1}{P_{fa}}\right)} - erfc^{-1}\left(2P_d\right) \right)^2 - \frac{1}{2} \right\} L} \right]^{1/4} . \tag{6.1}$$

The radar range equation in this form contains 11 variables. Two of the 11 variables are strictly fixed by system specifications: the probability of detection P_d and the maximum range to a target of RCS σ (given P_{fa} and P_d) $R_{\max;P_d,P_{fa},\sigma}$. As discussed in Section 5.3, the probability of pixel false alarm P_{fa} is determined by the specification of minimum average time between system false alarms. The average RCS σ is typically set by the choice of transmit frequency and the radar targets of interest. Among the remaining variable parameters are the antenna gains G_t and G_r, the transmit power P_t, the transmit wavelength λ, the receiver bandwidth B, the receiver noise figure F, and the aggregate system losses L.

An important stage of the radar design cycle is to enumerate the primary factors that influence parameter selection. Table 6.1 lists the main interdependencies among the parameters.

Table 6.1. Factors influencing radar parameter selection

Design Parameter	Symbol	Primary Influential Factors
transmit wavelength	λ	cost, Doppler resolution, FCC, size
baseband bandwidth	B	transmit wavelength, max target speed
transmit power	P_t	cost, power consumption, FCC, size
antenna gain	G_t, G_r	cost, angular coverage, size
receiver noise figure	F	cost, power consumption, size
system losses	L	cost, power consumption, size

For a specific design, several of the influencing factors of Table 6.1 may not be of high importance. Size and power consumption constraints were of minor concern to the radar design in this dissertation, while Doppler resolution and cost heavily influenced the design. In the next section, Table 6.1 will be further discussed as well as MFCW radar system design.

6.2 MFCW RADAR DESIGN

In this section, the design focus is narrowed to an MFCW radar used for collection of high Doppler resolution target signatures. Since Doppler resolution improves with increasing frequency, shorter transmit wavelengths are favored. However, obtaining inexpensive radar components past X-band (8-12GHz) is difficult. Since keeping cost low is a significant factor to this dissertation, a 3cm operating wavelength (10GHz frequency) was chosen as a tradeoff of cost and Doppler resolution. After selecting a transmit frequency, the receiver bandwidth is determined by the maximum expected target speed. If the maximum expected target speed is 35 meters per second (78 mph), the maximum Doppler frequency received (for a 10GHz transmit frequency) is approximately 2.35kHz. The chosen sampling rate is 5kHz, which is slightly higher than the Nyquist rate. The sampling rate then sets the baseband bandwidth to be roughly equal to twice the 3dB bandwidth of the anti-aliasing filters. For this dissertation, the complex baseband bandwidth is approximately 5kHz.

Cost, power, and size constraints typically dictate the selection of transmit power. In addition, the Federal Communications Commission (FCC) sets limits on the maximum radiated power density which factors into some high power budget designs. Cost is the main limitation on transmit power for this dissertation. Low cost surface-mount power amplifiers were chosen that offer over 16dBm (40mW) output power from a small, low power package.

For aperture-type antennas, the antenna size and gain are inversely proportional to the beamwidth. Thus, antenna selection is often dominated by angular coverage requirements and size (and/or weight) limits. Inexpensive, compact 17dB gain horn antennas were selected for this design. The antenna beamwidth is 25° in both the azimuth and elevation planes. Optimization of system noise figure and overall system losses within system requirements is a significant design challenge. In the next section, system design for reduced noise figure will be discussed.

6.3 SYSTEM DESIGN FOR REDUCED NOISE FIGURE

Noise figure refers to the reduction of signal to noise ratio from the input to the output of a device. One then defines the noise figure as

$$F = \frac{SNR_{in}}{SNR_{out}} \tag{6.2}$$

where SNR_{in} and SNR_{out} are the input and output signal to noise ratios, respectively. Implicit in the definition of noise figure is that the input noise to the device results from a matched (to the system characteristic impedance) resistor at a temperature of 290K. Receiver design relies heavily on noise figure optimization. To use noise figure in receiver design, one must determine the overall noise figure of the cascaded devices within the receiver. Cascading N devices results in an overall noise figure of

$$F = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} + \cdots + \frac{F_N - 1}{G_1 G_2 \cdots G_N} = F_1 + \sum_{i=2}^{N} \frac{F_i - 1}{\prod_{j=1}^{i-1} G_j} \tag{6.3}$$

where F_i is the noise figure and G_i is the gain of the i$^{\text{th}}$ device. Since the gain of previous stages reduces the effective noise figure of a device, the first few components of a receiver chain typically contribute the most to the overall system noise figure. Therefore, judicious placement of receiver front-end components can dramatically affect system noise levels.

The noise figures and gains of the front-end components used in this dissertation's design are listed in Table 6.2.

Table 6.2. Front-end component noise figures and gains

	Low Noise Amp	Power Divider	Balanced Mixer
Noise Figure	2.5 dB	3.7dB	8.5dB
Gain	15dB	-3.7dB	-8.5dB

In addition to the components of Table 6.2, assume a coaxial cable and connector loss at the input to the receiver of 0.5dB. This input loss constrains the first stage noise figure F_1 to be 0.5dB. For a 3FCW radar using quadrature demodulation, the received signal must be power-divided three times before reaching the mixers. The three power splits combine to give a rather large 11dB power splitter loss. In practice, power splitter loss limits the number of tones used in an MFCW radar. Figure 6.1 shows one possible receiver front end configuration.

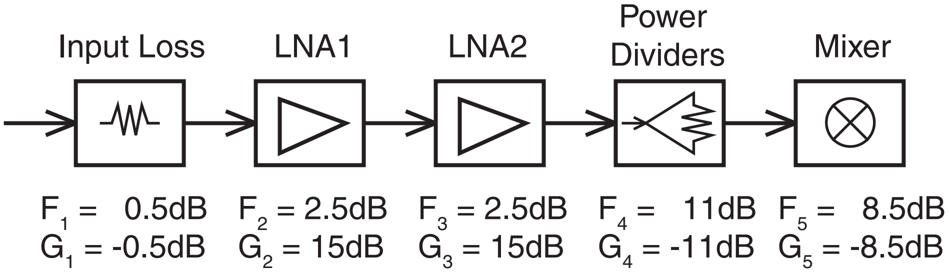


Figure 6.1. Receiver front end with two LNAs before the power dividers

The receiver front end of Figure 6.1 provides 10dB overall gain and 3.3dB noise figure. Other noise figures can be achieved by varying the position and number of low noise amplifiers (LNAs) as seen in Table 6.3.

Table 6.3. Varying receiver noise figure by changing number and position of LNAs

# of LNAs	LNA Position	Noise Figure	Gain
0	n/a	20 dB	-20 dB
1	after dividers	14.4 dB	-5 dB
1	before dividers	7.1 dB	-5 dB
2	after dividers	14.1 dB	10 dB
2	before dividers	3.3 dB	10 dB

Table 6.3 clearly shows the benefit of positioning the LNAs before the power dividers. Adding more LNAs improves the noise figure, but too many LNAs will deteriorate the receiver dynamic range along with cost and power budgets.

6.4 RADAR PARTS COST ANALYSIS

The system cost specification requires a per-unit parts cost of under $1,000. This cost point demanded significant design tradeoffs. For example, a single X-band power amplifier packaged in a shielded box with moderate performance can cost over $2,000. To reduce cost, a surface-mount power amplifier was chosen (with lower gain and output power) that costs under $30 per unit in small quantities. Table 6.4 lists the costs of various components of the radar design for small production quantities.

Table 6.4. Small quantity radar parts cost analysis

Cost Category	Aggregate Cost
Microwave components	$410
Cables and connectors	$100
Antennas	$40
Waveguide to coax converters	$200
Shielded chassis	$70
Microwave PCB fabrication	$180
Baseband PCB fabrication	$80
Baseband components	$165
Total	$1,245

Table 6.4 includes the cost of adding digital signal processor (DSP) hardware to perform the radar calculations instead of the personal computer currently being used. The radar parts cost of $1,245 nearly reaches the specified goal of less than $1,000 per unit even in small quantity (less than 10 units) production. The specified price point will likely be achieved for moderate to large production quantities. The prices listed in Table 6.4 are for the radar parts as it was designed in 2004. New receiver chips have been developed since 2004 that integrate the low noise amplifier, I/Q mixers, and local oscillator buffer into a single package with reduced cost.

Chapter 7: Database Collection and Pre-processing

7.1 DATA COLLECTION ENVIRONMENTS

Following the completion of the last radar hardware revision, an extensive database of radar signatures was collected from 2004 to 2007. Most of the radar tests were performed in an industrial environment in Austin, Texas as seen in Figure 7.1.

Figure 7.1. Industrial testing environment in Austin, Texas.

Several buildings, a concrete driveway, and parked vehicles all contributed to the complex electromagnetic propagation environment. Testing in this environment offered a more rigorous test than in an open field. The majority of human, vehicle, and dog datasets were collected in this environment. The goat datasets were taken at a central Texas ranch to minimize human-animal interaction. Figure 7.2 shows a picture of the environment.

Figure 7.2. Ranch in central Texas where goat datasets were collected.

The deer datasets were collected in a suburban neighborhood in Austin, Texas. As with the goat datasets, the site for collecting deer data was chosen to minimize human-animal interaction and obtain animal radar signatures with natural movement patterns. In addition to deer, data from passing vehicles was also collected at this site. A picture of the deer collection site is shown in Figure 7.3.

Figure 7.3. Deer data collection site in an Austin, Texas suburban neighborhood.

The bird data was collected in the backyard of a house in Austin, Texas. A large bird feeder attracted the birds to the test site. The radar was located inside a tent to prevent the birds from being spooked by the radar and/or the radar operator. Data was collected from multiple bird types in-flight and while walking on the ground. Figure 7.4 shows the bird data collection site.

Figure 7.4. Bird data collection site in an Austin, Texas suburban neighborhood

Figure 7.5 shows the mobile data collection platform built to facilitate the data collection process.

(a) (b)

Figure 7.5. Mobile data collection platform (a), and close-up view (b)

As seen in Figure 7.5, the transmit and receive horn antennas on the radar cart are mounted to a metal plate for rigidity and can be manually rotated on-axis. The metal plate's machined rotation stops accommodate any combination of linear polarizations. The radar hardware is encased in a metal enclosure directly behind the antenna fixture. The digital camcorder to the left of the radar unit provides audio-visual documentation of each data file collected. The computer to the right of the radar hardware stores the large amount of data acquired from simultaneous acquisition of radar signatures and digital video. The monitor behind the computer provides a real-time display of target spectrogram signatures and radar parameters (target range, velocity, etc.). The mobile data collection platform transports easily to the testing location of choice.

7.2 DATABASE COLLECTION

The goal to thoroughly study the micro-Doppler phenomenon required the collection of an expansive radar signature database. During the course of this dissertation work, over 650 MFCW radar data collection runs were performed. These 650 data collection runs were then processed and labeled into over 3,900 classification frames of 2.46 second duration (as described in Chapter 5). The goal throughout the dissertation was to obtain a wide variety of data files that represent operational conditions as much as possible. For example, targets approached the radar at numerous angles of approach, not simply directly toward the radar. Data files were also taken of targets moving randomly in the radar range. The acquired data included three main target classes: human, vehicle, and animal. Each target sub-class contains data from targets with varying speeds, approach angles, and ranges. The human dataset includes files where the individual crawls toward the radar. The crawling files include hands-and-knees baby crawling in addition to army crawling. The vehicle datasets include the signatures from a sports utility vehicle and a station wagon. However, vehicles randomly passing through the radar range also became part of the database. The animal dataset contains data from deer, birds, dogs, and goats. The animal motion was unconstrained, thus presenting the classifiers with realistic animal micro-Doppler signatures. The contents of the micro-Doppler signature database will be discussed in greater detail in Chapter 12.

7.3 DATABASE PRE-PROCESSING

Figure 7.6 illustrates the software pre-processing steps prior to joint time-frequency detection for a 3FCW radar.

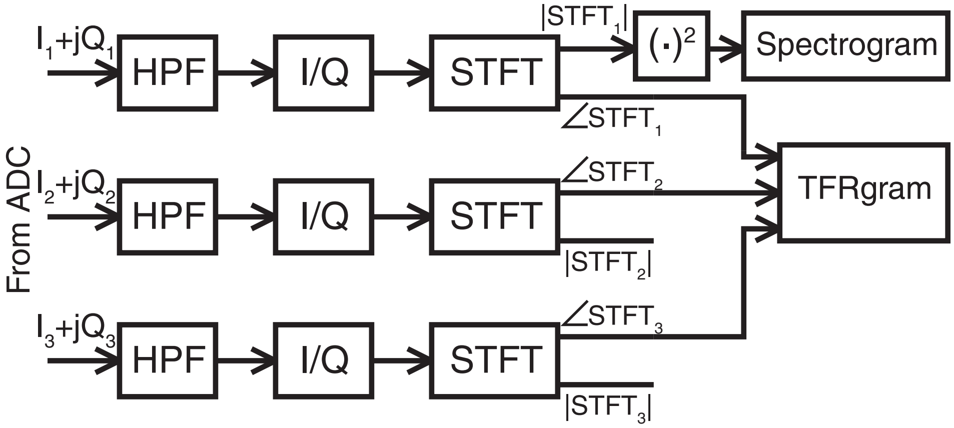


Figure 7.6. 3FCW radar pre-processing prior to joint time-frequency detection

For the MFCW radar, analog to digital conversion produces M complex baseband digital signals where M is the number of transmitted tones. A high-pass filter processes each complex channel to remove clutter and 1/f noise around DC. The filter stopband must be narrow to minimize unwanted attenuation of target signal energy near DC. The high-pass filter magnitude response is shown in Figure 7.7.

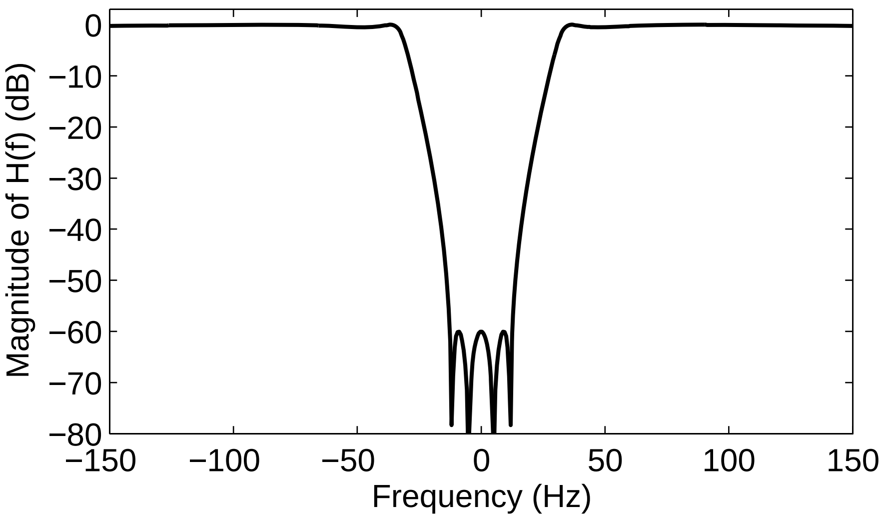
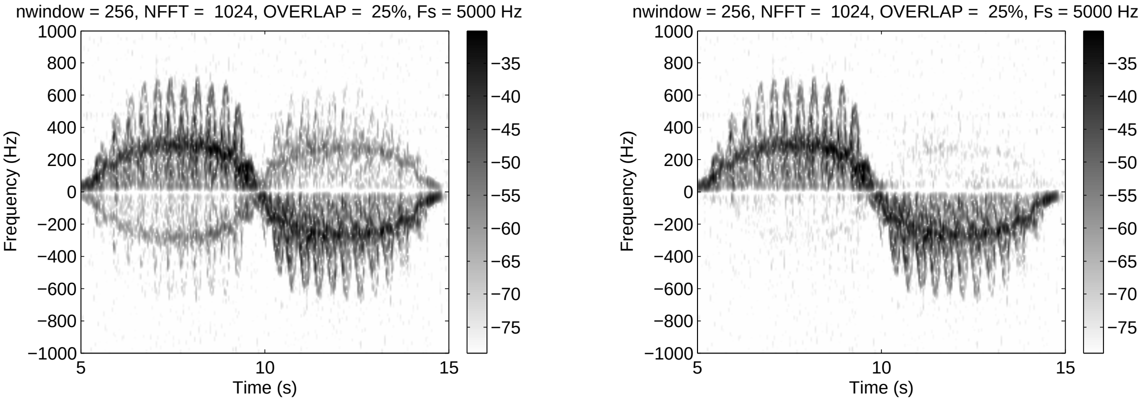


Figure 7.7. Magnitude of fourth-order elliptic high-pass filter transfer function

The filter is a fourth-order elliptic high-pass filter with a 31.5Hz cutoff frequency, 60dB stopband attenuation, and 0.5dB ripple in the passband. The phase response is non-linear (the elliptic filter is IIR), but using the same filter on all complex channels maintains the proper phase difference between channels.

Following filtering, each inphase/quadrature signal pair undergoes I/Q error correction. I/Q error results from amplitude and/or phase imbalance in the quadrature demodulator or mismatched receiver channels following quadrature demodulation. I/Q error creates an “image target” in the spectrogram at the negative of the true target frequency. Providing that the image target is strong enough, a target tracking algorithm may report two targets present when there is actually only one. Figure 7.8 (a) shows a spectrogram of a human walking toward the radar and then turning around and walking away.



(a) (b)

Figure 7.8. Spectrogram with strong image target (a), and corrected spectrogram (b)

An amplitude imbalance of 0.5mV and a phase imbalance of 14º produced an image target approximately 18dB below the true target power. Applying an I/Q error correction algorithm [40] reduced the image target power as seen in Figure 7.8 (b).

After I/Q error correction, the short-time Fourier transform (STFT) converts each complex channel into the joint time-frequency domain. The magnitude squared of one complex channel's STFT forms the spectrogram for target detection and tracking as developed in Section 5.3. The phase of the STFT of each channel is used to calculate the TFRgram as described in Section 5.4. The STFT is also utilized for feature extraction as discussed in Chapter 9.

Chapter 8: Detection Performance and Micro-Doppler Analysis

8.1 DETECTION PERFORMANCE

An accurate analysis of a system's detection performance is often difficult to obtain for several reasons. Predicting performance under operational conditions adds additional complexity due to wide variations in target and noise statistics. Some of the difficulty arises due to RCS and multipath-induced signal fluctuation. The theory developed in Chapter 5 provides acceptable results for targets of constant or slowly varying amplitude. Both multipath and RCS variations often produce rapidly varying target amplitudes which result in brief periods of target loss. Another difficult task is calculating an accurate system false alarm rate. With average false alarm times specified in days, large amounts of data are required to determine the false alarm rate. If automated methods of data collection are used, verifying a true false alarm also becomes challenging. This dissertation will focus on predicted maximum target range for a given (theoretical) probability of false alarm and detection and compare results to experimental data.

Using equation (6.1), the predicted maximum range for various targets is tabulated in Table 8.1.

Table 8.1. Maximum detectable target ranges for $P_d = 0.9$ and $P_{fa} = 3.3 \times 10^{-7}$

Target	Average RCS	Maximum detectable range
Human	0.75m^2	73.8m (242ft)
Vehicle	10m^2	141m (463ft)
Animal	0.1m^2	44.5m (146ft)

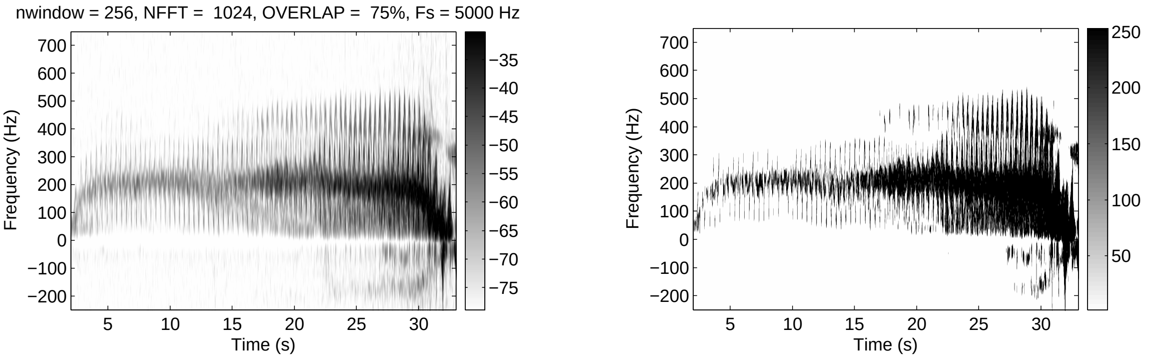
Note that Table 8.1 utilizes a single, constant value for the RCS of each class. In practice, signal fading will reduce the maximum detectable range. The RCS values in Table 8.1 are (average) approximations for the targets of interest to this dissertation.

The predicted results of Table 8.1 will now be compared to the results from experimental data. As discussed in Section 5.3, a *system-level* detection is defined as an event when two or more *pixel* detections occur within a single FFT. The empirical maximum detectable range to a target is sought for a specified system-level probability of false alarm and detection. The following empirical calculation of the maximum detectable range is utilized in this dissertation. If, for example, the time between spectrogram FFTs is 38.5ms, the number of FFTs per second is approximately 26. For a one second observation period and a probability of system-level detection of 0.9, the average number of missed detections is approximately three at the maximum detectable range ($26 \times 0.1 \times 1 = 2.6 \approx 3$). The maximum detectable range is then the shortest range where the number of missed detections over a one second interval is three or more. More generally, for an FFT time interval T_{FFT} and a system-level probability of detection $P_{d,sys}$

$$N_{miss} \approx \left\lceil \frac{1}{T_{FFT}} (1 - P_{d,sys}) \Delta t \right\rceil \tag{8.1}$$

where $\lceil \bullet \rceil$ is the ceiling operation and N_{miss} is the average number of missed detections over the interval Δt.

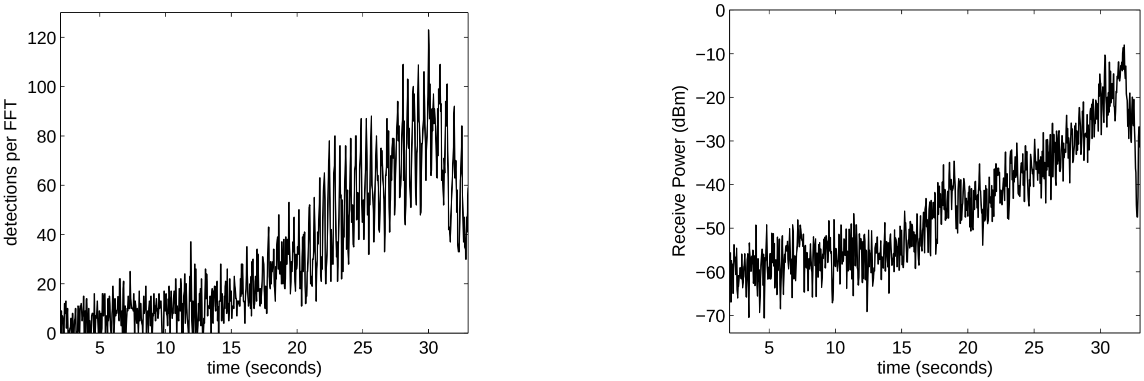
Figure 8.1 shows the spectrogram of a human jogging from 292ft to 15ft before and after detection. From the post-detection spectrogram, system-level detections (two or more detections per FFT) are achieved out to 292ft. However, there are many time intervals where no system-level detections occur.



(a) (b)

Figure 8.1. Spectrogram of jogging human before (a), and after (b) detection

The number of detections per FFT from Figure 8.1 is calculated and displayed in Figure 8.2 (a). Figure 8.2 (b) plots the received power versus time for the jogging human.



(a) (b)

Figure 8.2. Number of detections per FFT (a), and received power (b) for jogging human

The fast, deep signal fades that occur in typical datasets are readily seen. These signal fades result in higher numbers of missed detections in regions of low SNR.

Analysis of the data of Figure 8.2 (a) results in three missed detections per second occurring around 8.7 seconds. The target's range at 8.7 seconds was 71 meters. Thus, the approximate maximum detectable range for this target is 71 meters. The results of further detection range testing are listed in Table 8.2.

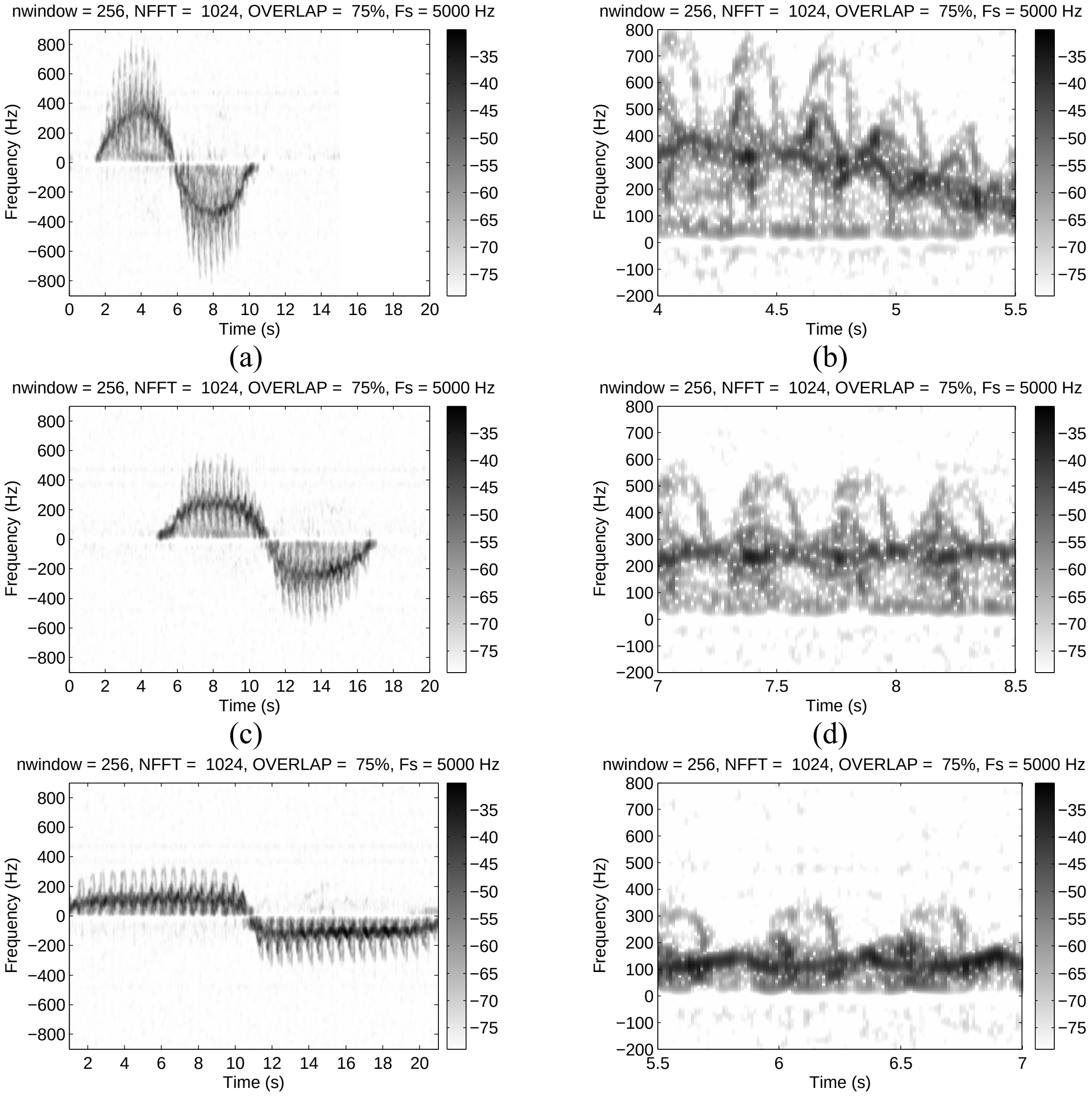
Table 8.2. Empirical maximum detectable range

Target Class	Average maximum detectable range	# of files averaged
Human	66m (218ft)	5
Vehicle	105m (345ft)	5
Animal	40m (131ft)	5

Five files in each target class were averaged to obtain these results. Comparing Table 8.2 to Table 8.1, the observed maximum ranges were less than the predicted maximum ranges in all target classes. A primary cause of this discrepancy is the deep signal fading shown in Figure 8.2 (b).

8.2 MICRO-DOPPLER ANALYSIS

Figure 8.3 shows spectrograms of a human target running, jogging, and walking from a range of 150ft to 100ft and then back to 150ft.

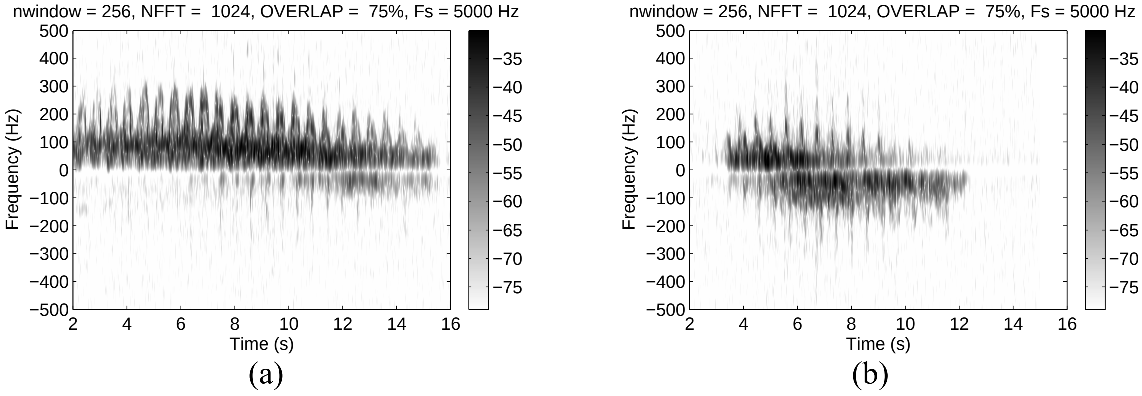


(a) (b) (c) (d) (e) (f)

Figure 8.3. Spectrogram of human running (a), jogging (c), and walking (e). Expanded views are (b), (d), and (f), respectively

Figure 8.3 (a), (c), and (e) reveal the coarse differences in micro-Doppler over varying target speed. As expected, maximum micro-Doppler frequency and leg-swing rate increase as the target reaches faster speeds. The expanded views in Figure 8.3 (b), (d), and (f) expose fine micro-Doppler features. Returns from all target scattering centers that move at nearly the same speed as the torso, “torso returns,” appear as the darkened line running through the spectrograms. The large arches over the torso return result from leg-swing motion. When one leg is planted on the ground (foot not moving relative to the radar), that leg’s return appears at DC. Arm-swing can be seen as the smaller arches around the torso return. As seen in Figure 8.3 (b), (d), and (f), the arm-swing becomes more pronounced with higher speed.

The targets in Figure 8.3 all move along a radial path from the radar that is centered on the radar antennas. This zero degree approach angle makes the relative velocity calculated from the Doppler return (approximately) the true target velocity. Data files were acquired from targets moving at various approach angles to observe its effect on the radar signatures. Figure 8.4 (a) shows the spectrogram of a human walking at a 45º approach angle from a range of about 100ft to 50ft.



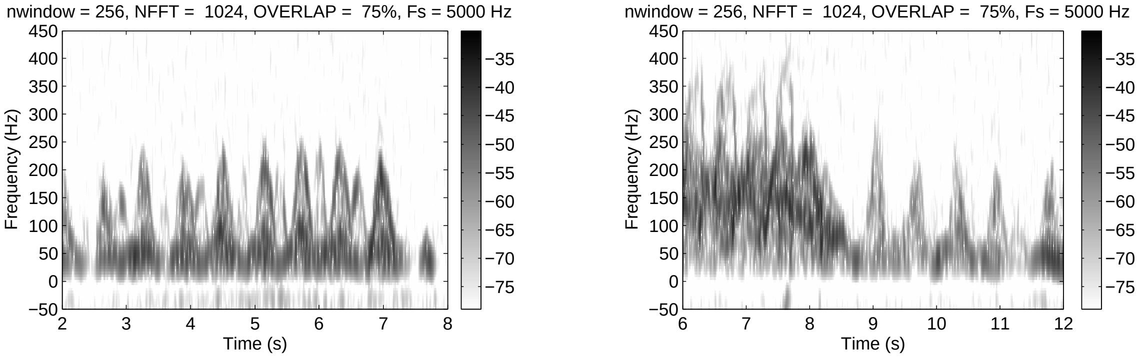
(a) (b)

Figure 8.4. Human target walking at 45º (a) and 90º (b) relative to radar

The spectrogram of a target approaching at 45º does not drastically differ from that of a target approaching at 0º. However, the received signal strength appears to drop off near the end of the file, even though the target is approaching the radar. This effect is explained by the fact that the target is passing through the antenna beamwidth. Figure

8.4 (b) shows a human walking at (approximately) a 90º angle relative to the radar at 50ft range. The radar still receives a strong return from the target, but the spectrogram looks quite different from the 0º approach angle case. The target signature is now centered on DC and the returns from leg and arm motion extend in both positive and negative frequency directions. In addition, the leg and arm returns are of much shorter time duration.

Crawling human targets illustrate the effects of varying posture on target signatures. Figure 8.5 (a) shows a human crawling toward the radar from a range of 100ft to 85ft.

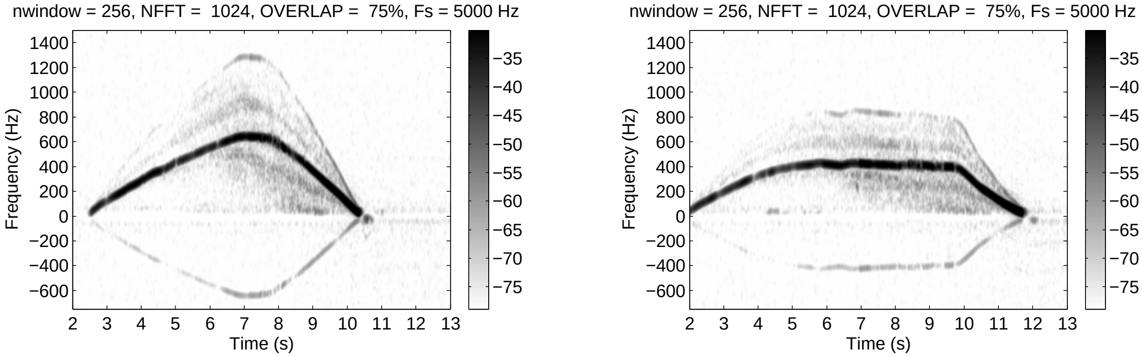


(a) (b)

Figure 8.5. Human crawling (a). Transition from walking to crawling (b).

The crawling target's torso return concentrates close to DC. In contrast to a walking human, the crawler's micro-Doppler energy from leg and arm motion is highly skewed to one side of the torso return. In addition, the period between micro-Doppler peaks from the limbs is typically longer for a crawler than for a walker. This effect is illustrated in Figure 8.5 (b), which shows a human transitioning from walking to crawling.

In addition to humans, data was collected from vehicular targets. Most of the vehicle data came from either a sports utility vehicle or a station wagon, but random passing vehicles also became part of the datasets. Major differences exist between human and automobile Doppler signatures. As seen in Figure 8.6, a typical automobile radar signature is strongly concentrated on a single line in the time-frequency plane. Figure 8.6 displays the spectrograms of an automobile driving from a range of 250ft to 100ft with various accelerations and velocities.



(a) (b)

Figure 8.6. Vehicle driving at different velocities and accelerations

Most vehicles contain several predominant scattering centers, but all of these scattering centers move at roughly the same speed relative to the radar. A notable exception is a helicopter where the rotor blades cause large sinusoidal micro-Doppler around the helicopter body return [20]. Since most automobiles are rigid scatterers, one would not expect to see (in theory) the Doppler returns in Figure 8.6 (a) other than the main return. An image signal can be seen at the negative frequency of the main return. This signal is not a physical phenomenon of the target but an artifact of receiver inphase/quadrature error (as discussed in Section 7.3). An I/Q error correction algorithm was applied to this data, but complete error correction was not achieved. In addition to the I/Q error signal, another artifact signal occurs at twice the frequency of the main return. This artifact signal results from harmonic distortion in the receiver at high input signal levels. Figure

8.6 (a) also contains artifacts from the phase noise of the local oscillators. Phase noise artifacts can be seen close-in on either side of the main return. Artifact signals are present in human data also, but the signal strength of vehicular targets raise the artifact signal levels well above the system noise floor.

As with the human targets, vehicle approach angle was varied to observe its effect on the Doppler signatures. Figure 8.7 (a) shows the spectrogram of a vehicle driving from a range of 250ft to 150ft at an angle of approximately 45°.

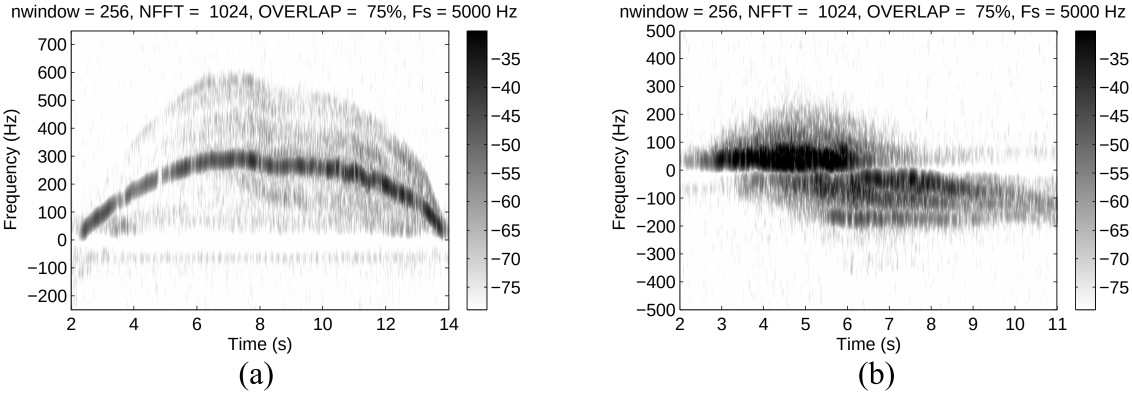


Figure 8.7. Vehicle driving at 45º (a) and 90º (b) relative to radar

The vehicle’s Doppler signature does not change significantly by varying the approach angle from 0° to 45°. However, when the approach angle is approximately 90°, the vehicle Doppler signature differs markedly as seen in Figure 8.7 (b). The primary difference is that the main return energy is spread out much more in frequency for the 90° approach angle case. Comparing Figure 8.7 (b) to Figure 8.4 (b), reveals that at a 90° approach angle, distinguishing between vehicle and human spectrograms becomes more difficult.

Figure 8.8 displays an interesting spectrogram of a vehicle turning a corner at close range to the radar.

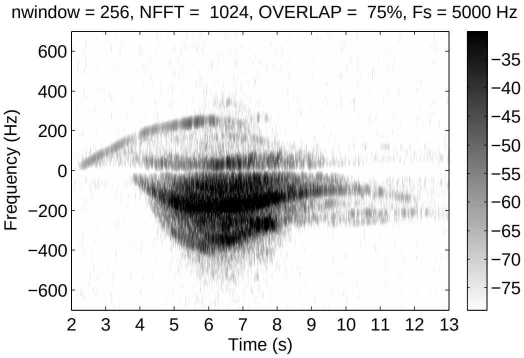


Figure 8.8. Vehicle turning in front of radar at close range

Referring back to Figure 7.1, the vehicle begins by driving behind and to the left of the radar unit. The vehicle then takes a right turn in front of the radar and continues to drive until the vehicle is well beyond the antenna beamwidth. One interesting feature of Figure 8.8 is the positive frequency return that starts at around 2 seconds and ends at 6 seconds. This return is not an artifact but a multipath return from the building in the upper right-hand corner of Figure 7.1. The multipath return shows up almost 2 seconds before the main return, and a tracking algorithm would likely report the multipath signal as another target.

Figure 8.9 shows the spectrogram of a vehicle driving toward the radar at a 0° approach angle and coming to a complete stop 50ft in front of the radar.

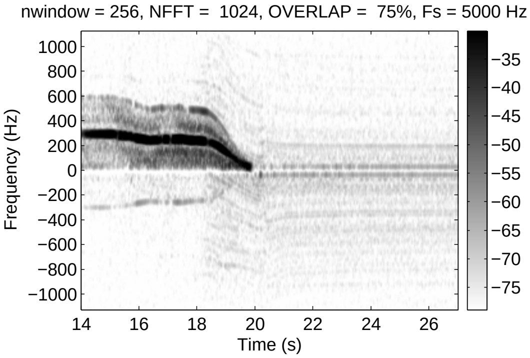
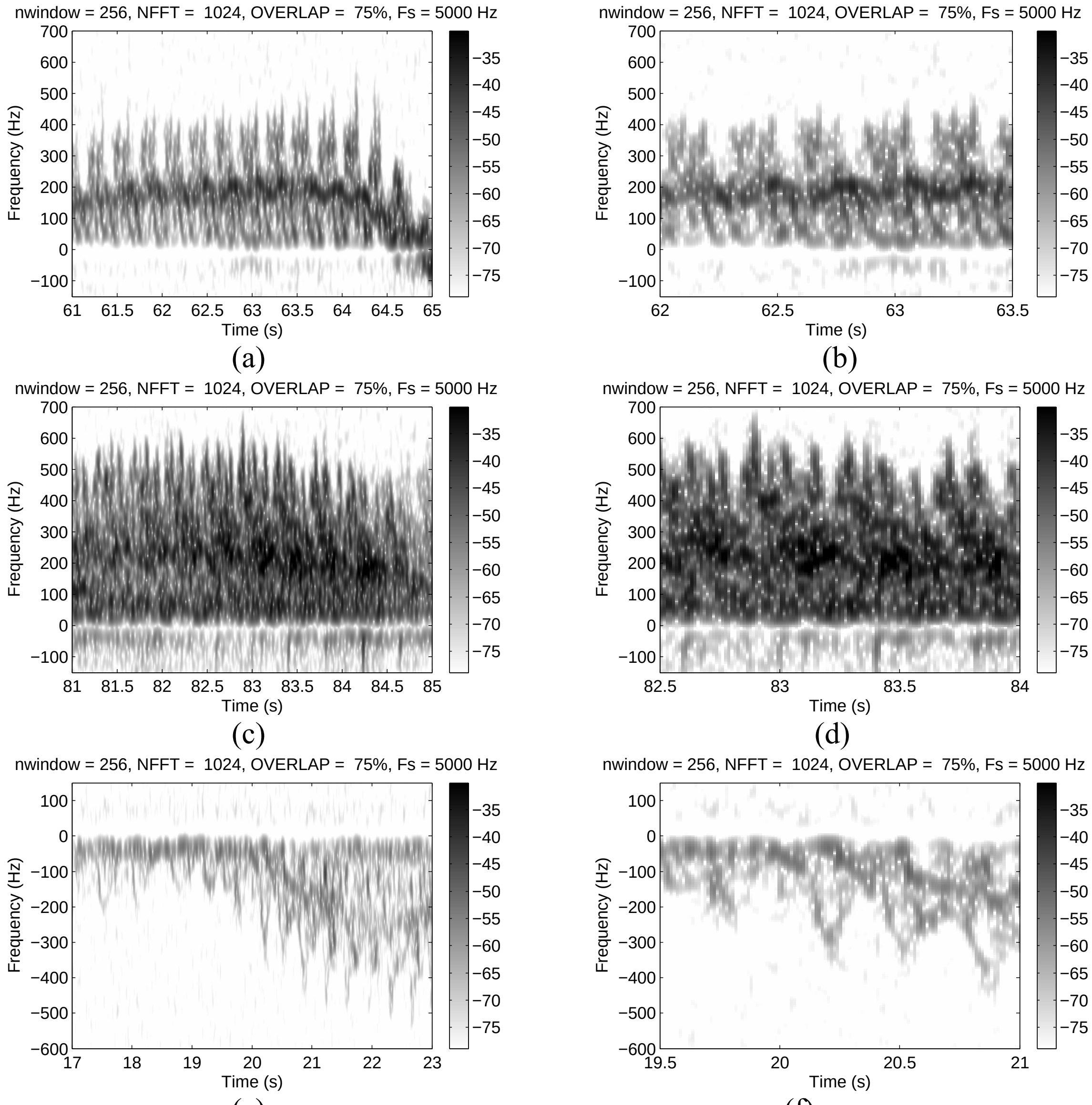


Figure 8.9. Micro-Doppler produced by vehicle engine is visible at close range

Although the vehicle is at a complete stop, the spectrogram contains numerous micro-Doppler lines. A combination of vibrating and rotating parts in the vehicle's idling engine produced the various micro-Doppler features of Figure 8.9.

In addition to humans and vehicles, the micro-Doppler features of animals were also studied. The animal datasets included dog, goat, deer, and bird. Figure 8.10 displays spectrograms of the first three animal types.



(a) (b)

(c) (d)

(e) (f)

Figure 8.10. Spectrogram of dog (a), goat (c), and deer (e). Expanded views are (b), (d), and (f), respectively

A dog approaching the radar produced the spectrogram of Figure 8.10 (a). Four goats running toward the radar contributed to the Doppler signature of Figure 8.10 (c). In Figure 8.10 (e), a deer casually wanders around until it leaps into the air and bolts off after becoming startled. Animal leg motion produces micro-Doppler of short time duration compared to a human approaching the radar at the same speed. This effect is due to the observation that most quadruped animals' stride rate is considerably faster than for biped animals. The overall micro-Doppler signature due to dogs and goats appears fairly similar. The deer micro-Doppler (other than a faster stride rate) closely resembles the micro-Doppler of a crawling human as seen in Figure 8.5. Before the deer bolts off, the main return from the deer is near DC, with the leg micro-Doppler skewed to one side, similar to the return from a crawling human.

In addition to land animals, bird micro-Doppler signatures were also collected. Figure 8.11 is a micro-Doppler signature collected from a dove.

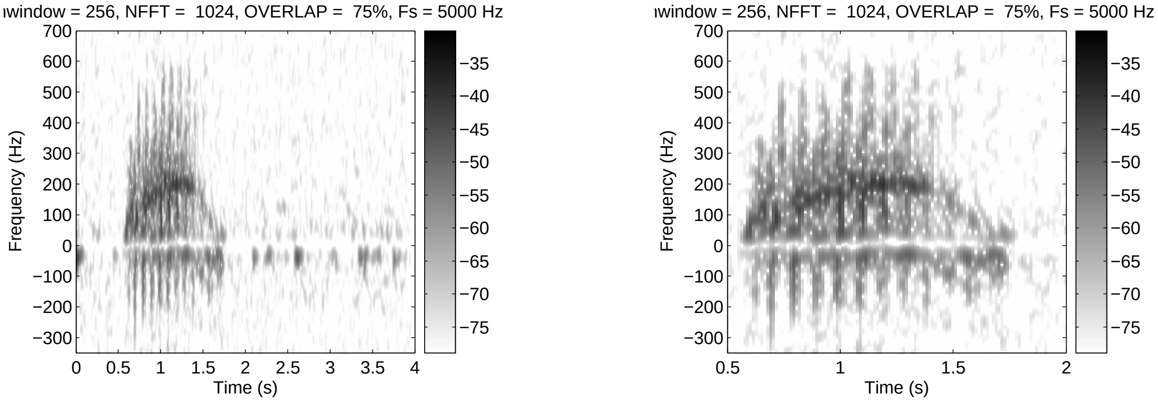
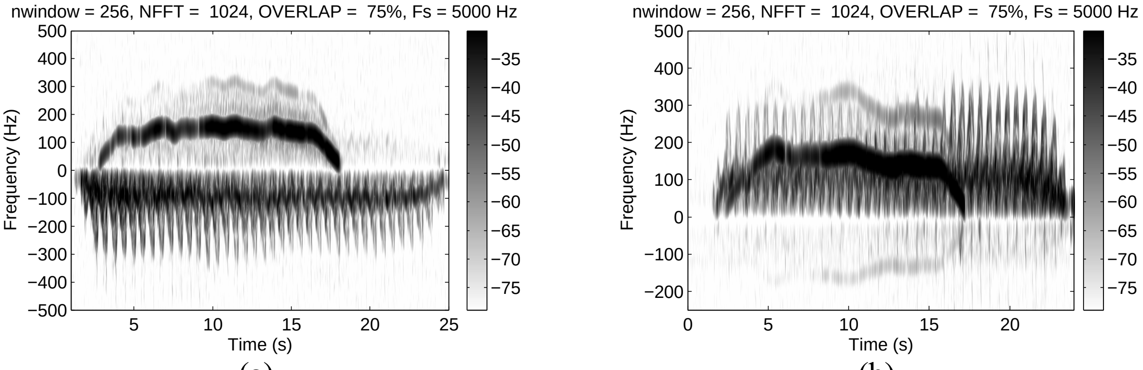


Figure 8.11. Spectrogram of bird (a) and expanded view (b)

The bird is initially on the ground feeding, and then it flies toward the radar. The returns seen from about 2 seconds to about 4 seconds are from other birds feeding on the ground. Note that the narrowband notch at DC is an artifact of preprocessing (high-pass filtering), and it is not part of the actual bird micro-Doppler response. Figure 8.11 reveals several interesting features of the bird micro-Doppler response. The high rate of the bird's wings flapping causes a short micro-Doppler period and high Doppler frequency relative to the torso. As seen in Figure 8.11 (b), the high rate of wing flapping is nearing the limits of the STFT's time-frequency resolution. Another feature of bird micro-Doppler is the high symmetry of the wing-produced micro-Doppler about the torso. A land animal must plant its feet to produce locomotion while a bird's wing motion is much less restricted. A land animal planting its feet during locomotion causes periodic micro-Doppler returns with zero velocity relative to the radar. This feature helps distinguish flying bird micro-Doppler from land animal micro-Doppler.

All of the previously displayed datasets contain only one target in each data file. Data files were also collected with multiple targets in motion at the same time. Two examples of multi-target data files are shown in Figure 8.12.

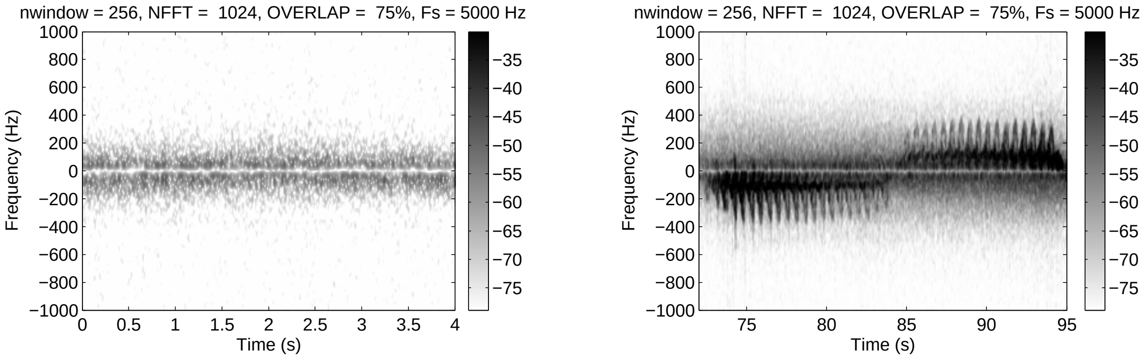


(a) (b)

Figure 8.12. Multiple targets moving in opposite directions (a), and the same direction (b)

Figure 8.12 (a) displays the spectrogram of a vehicle driving from 250ft to 150ft while a human walks from 50ft to 150ft. With the targets moving in opposite directions, little Doppler overlap occurs. The individual targets can be isolated by their Doppler separation only. In Figure 8.12 (b), a human walks from 150ft to 50ft while a vehicle drives from 250ft to 150ft. Since the two targets move in the same direction and at roughly the same speed, significant Doppler overlap occurs.

Some data was also acquired during adverse weather conditions. The spectrogram of ambient conditions during heavy rain appears in Figure 8.13 (a).



(a) (b)

Figure 8.13. Spectrogram of ambient conditions (a), and human target (b) during heavy rain

A significant clutter region forms around DC due to the rainfall. The rest of the spectrogram is relatively unaffected by the rain. Figure 8.13 (b) shows a human walking away from and then toward the radar during heavy rain. The target is still easy to detect visually, but a fixed-threshold automatic detection algorithm would likely report a high number of false alarms due to the rain. A variable-threshold detection algorithm is required to control the false alarm rate in adverse weather conditions.

Chapter 9: Classifier Design and Statistical Analysis

9.1 INTRODUCTION

Classification is the process of separating objects into groups by comparing their attributes. The human body is an example of a complex classification system. Our eyes, skin, tongue, ears and nose constantly deliver raw information to our brains about the world around us. The brain is tasked with sorting through these raw signals and extracting important features for a given classification task. For example, while driving our brains use our sense of sight to estimate the size and shape (among other features) of road obstacles. If a child were to walk into the middle of the road, the brain would use the extracted features to (hopefully) classify the object in the road as a child and stop the vehicle. The massive parallelism of the brain allows humans to perform complex classification tasks with ease.

Due to the ability of the human brain to excel at classification under widely varying conditions, many classification systems today still incorporate a human operator at some point in the system. In fully-automatic classification systems, a computer performs all steps of the classification procedure with no assistance from a human operator. This dissertation involves the design of fully-automatic target classifiers. Figure 9.1 illustrates the procedural block diagram of a typical classification system.

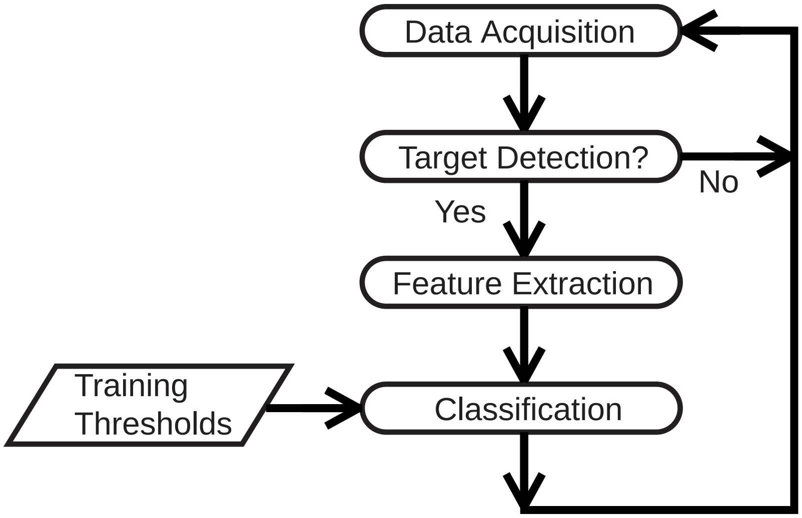


Figure 9.1. Block diagram of a typical real-time classification system

The classification procedure begins with the acquisition of a frame of raw data. Basic pre-processing steps such as high pass filtering are performed in this stage. The data are then fed into a target detection algorithm. If the detection algorithm decides that a target is present, the target features are extracted from the data. In the feature extraction stage, the data frame is processed to quantify the various target features used for classification. In the training phase, feature selection would follow feature extraction. Feature selection seeks to remove any highly correlated features (features containing redundant information) from the extracted feature set. After feature extraction, the classification stage utilizes statistical models and thresholds created during the training procedure for discrimination. The classification step produces the estimated target class, and then the cycle repeats for subsequent data frames.

Data acquisition and target detection were discussed in earlier sections of this dissertation. Feature extraction, feature selection, classifier training, and classifier performance evaluation will be the topics of the remainder of this dissertation.

9.2 STATISTICAL FEATURE EXTRACTION

Obtaining robust feature vectors through the process of feature extraction and selection is vital to the design of any classification system. The chosen feature set must maintain reasonable class separability under adverse conditions (low SNR, high interference, background variation, system configuration changes, etc.) to be of practical use. Unfortunately, there is no single feature set that performs well (or is appropriate) for all classification tasks. In contrast to statistical learning theory, the mathematical framework of feature extraction and selection is quite limited. The majority of feature extraction and selection algorithms in practice today are based on ad-hoc, heuristic methods.

Chapter 8 highlighted the important role of joint time-frequency processing in (visual) target classification. One could potentially use the time-frequency (spectrogram) coefficients directly as a feature set. Although spectrograms contain fine detail that is useful for visual classification, they fail to provide a compact data representation for efficient computation. A typical spectrogram-based feature vector might contain around 16,000 elements—far too large for small or even moderately large training sets to appropriately model the resulting high-dimensional feature space.

One feature set that has been proven quite effective by the speech recognition community is the cepstrum coefficients [50]. The discrete-time real cepstrum $c[n]$ is defined as the inverse discrete Fourier transform of the log-magnitude spectrum of sequence $x[n]$

$$c[n] = \mathrm{IDFT}\left[\log\left|\mathrm{DFT}\left(x[n]\right)\right|\right]. \tag{9.1}$$

The real cepstrum efficiently codes the spectral information of a signal, similar to the use of the discrete cosine transform (DCT) in image compression. Filter-bank methods and linear predictive coding (LPC) are the two primary approaches to calculating the cepstral coefficients. Filter-bank methods directly implement equation (9.1) by decomposing the spectrum into a set of overlapping filters. Linear predictive coding [50] is an all-pole system that models a signal $x[n]$ as the linear combination of the previous p samples

$$\hat{x}[n] \cong \sum_{k=1}^{p} a_k x[n-k] \tag{9.2}$$

where a_k are the LPC coefficients. The residual error in the LPC estimate is defined as follows

$$e[n] = x[n] - \hat{x}[n] = x[n] - \sum_{k=1}^{p} a_k x[n-k] \tag{9.3}$$

where the mean-squared error is

$$E_{LPC,\,residual} = \sum_{n} e^2[n] = \sum_{n} \left[x[n] - \sum_{k=1}^{p} a_k x[n-k] \right]^2 . \tag{9.4}$$

The mean-squared LPC error (residual energy) is often used as a feature for classification. The linear prediction cepstral coefficients (LPCC) are derived from the LPC coefficients through the following recursion relation [50]

$$c_k = -a_k - \frac{1}{k} \sum_{m=1}^{k-1} m c_m a_{k-m} \quad 1 \le k \le p . \tag{9.5}$$

Due to the non-stationary nature of the radar signal, the LPCCs must be re-estimated frequently to avoid inaccurate parameter values. The LPCCs are typically estimated at the same rate as the FFT spectral analysis (e.g., every 38.5ms).

The cepstral coefficients of equation (9.5) are often referred to as the static cepstral coefficients. The first and second temporal derivatives of the cepstral coefficients are known as the delta and delta-delta cepstral coefficients, respectively. The dynamic cepstral coefficients incorporate spectral slope and curvature information. Adding dynamic cepstral coefficients to the set of static cepstral coefficients has been shown to improve classification performance [51], [52]. Figure 9.2 illustrates the cepstral processing procedure used in this dissertation.

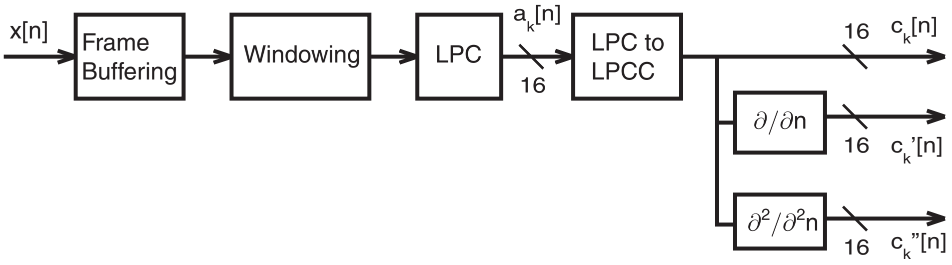


Figure 9.2. LPCC feature extraction

In theory, the cepstral derivatives can be calculated through a simple first-order finite difference such as $\frac{\partial c_k[n]}{\partial n} \cong c_k[n] - c_k[n-1]$. However, the first order finite difference is noisy in practice. To alleviate this problem, the cepstral derivatives are calculated by differentiating a (sliding-window) second-order polynomial fit to the data. The first and second temporal cepstral derivatives are [50]

$$c_k^{'}[n] = \frac{\partial c_k[n]}{\partial n} \cong \sum_{\tau=-M}^{M} \tau c_k[n+\tau] \Big/ T_M \tag{9.6}$$

and

$$c_k^{''}[n] = \frac{\partial^2 c_k[n]}{\partial^2 n} \cong \frac{2\left\{ T_M \left[\sum_{\tau=-M}^{M} c_k[n+\tau] \right] - (2M+1) \left[\sum_{\tau=-M}^{M} \tau^2 c_k[n+\tau] \right] \right\}}{T_M^2 - (2M+1) \left[\sum_{\tau=-M}^{M} \tau^4 \right]} \tag{9.7}$$

where n is the discrete-time index, 2M+1 is the width of the sliding window, and $T_M = \sum_{\tau=-M}^{M} \tau^2$. A sliding window length of 7 ($M = 3$) was used in this dissertation to provide adequate smoothing of the derivative estimates.

Figure 9.3 shows the c_1 cepstral coefficient time-series overlaid on a spectrogram of a human running. Note that the y-axis (frequency) is only applicable to the spectrogram (the cepstral coefficient has been scaled for easier visual comparison).

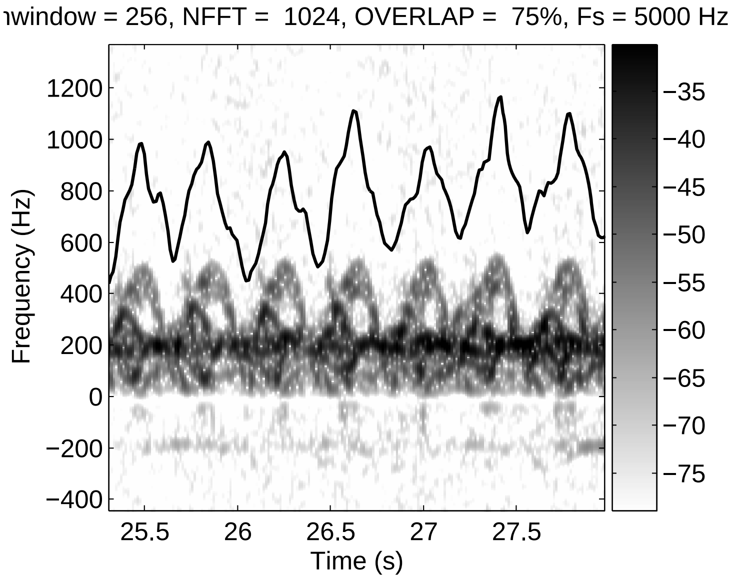


Figure 9.3. Cepstral coefficient c_1 overlaid on a spectrogram of a human running.

The cepstral coefficients efficiently extract spectral periodicity. For comparison, the spectrogram of Figure 9.3 requires 128 kilobytes (using double precision floating point) to store while the 16 cepstral coefficients along with their first and second order derivatives require 24 kilobytes (a 5.3:1 compression ratio). Efficient dimensionality reduction is vital to achieve acceptable classification performance when using limited-size training sets.

In [53], the authors show that the energy of the cepstral coefficients and the energy of the cepstral derivatives are good classification features. The cepstral energy, delta-cepstral energy, and delta-delta cepstral energy are defined as follows

$$E_{cepstral}[n] = \sum_k \left(c_k[n]\right)^2 \tag{9.8}$$

$$E_{delta-cepstral}[n] = \sum_k \left(c_k^{'}[n]\right)^2 \tag{9.9}$$

$$E_{delta-delta-cepstral}[n] = \sum_k \left(c_k^{''}[n]\right)^2 . \tag{9.10}$$

As shown in [53], the energy of the cepstral coefficients contains much of the information-bearing content of the cepstral coefficients.

9.3 HEURISTIC FEATURE EXTRACTION

Heuristic features incorporate application-specific knowledge of the given classification task. Reliance on domain-specific knowledge can limit the applicability of heuristic features. In contrast, statistically based features such as the cepstral coefficients are often applicable to a much wider variety of pattern recognition problems. Heuristic features are typically derived via ad-hoc methods and often do not have strong mathematical or theoretical basis. However, heuristic features (based on expert knowledge of the physical phenomena underlying a classification problem) often greatly improve performance. For the classification problem presented by this dissertation, the underlying physical phenomenon is micro-Doppler. Thus, the majority of heuristic features used in this dissertation are derived from target micro-Doppler.

Heuristic feature extraction begins with the estimation of basic target parameters such as received signal power and target range. Target features used for classification are derived from the basic target parameters. Estimates of a target's micro-Doppler period, micro-Doppler energy, and range-weighted target energy are particularly useful classification features.

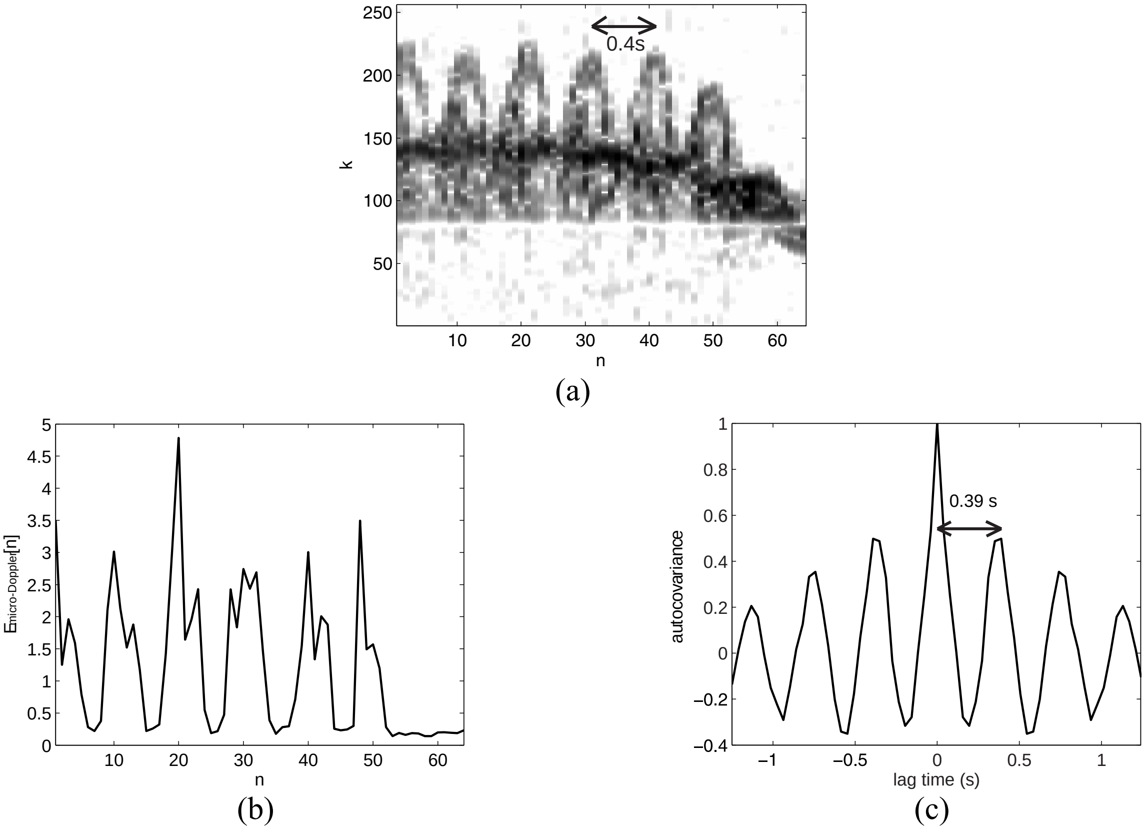
An estimate of a target's micro-Doppler period is an important classification feature. This feature calculates the period of non-rigid-body scattering centers that produce the micro-Doppler effect. The micro-Doppler period feature corresponds physically to the motion of limbs from human and animal targets. Calculating the micro-Doppler period starts with accumulating the target energy above (or below, depending on the direction of motion) the bulk-scatterer (e.g., torso) return

$$E_{micro-Doppler}[n] = \sum_{k > k_{bulk} + k_0} \left| STFT[k,n] \right|^2 \tag{9.11}$$

where k_{bulk} is the center frequency bin of the bulk-scatterer return, and $E_{micro-Doppler}[n]$ is the target micro-Doppler energy (another feature useful for classification). The fixed offset k_0 removes bulk-scatterer energy from the micro-Doppler energy estimate. The discrete autocovariance function $c_{xx}(\tau)$ provides a measure of the period of a discrete-time sequence $E_{micro-Doppler}[n]$. The autocovariance function is defined as

$$\begin{gathered} c_{xx}(\tau) = \frac{1}{M} \sum_{i=1}^{M-\tau} (E_{micro-Doppler}[i] - \hat{\bar{E}}_{micro-Doppler})(E_{micro-Doppler}[i+\tau] - \hat{\bar{E}}_{micro-Doppler}) \\ \text{where } \hat{\bar{E}}_{micro-Doppler} = \frac{1}{M} \sum_{i=1}^{M} E_{micro-Doppler}[i] \text{ and } \tau = 0,1,\ldots,M \end{gathered} \tag{9.12}$$

where M is the number of samples and τ is the lag variable. The autocovariance function is typically normalized so that the autocovariance at zero lag is equal to unity. Peaks in the autocovariance function reveal periodicities in the input sequence. The micro-Doppler period estimate $T_{micro-Doppler}$ is then the time difference between the peak at zero lag and the next closest peak of $c_{xx}(\tau)$. Figure 9.4 shows calculation of the micro-Doppler period for a human target.



(a)

(b)

(c)

Figure 9.4. Calculating micro-Doppler period: spectrogram (a), micro-Doppler energy $E_{micro-Doppler}[n]$ (b), and autocovariance (c)

Figure 9.4 indicates that the autocovariance provides a good estimate of average micro-Doppler period. The micro-Doppler period helps distinguish biped (human) and quadruped animals. The micro-Doppler period estimate is particularly useful when attempting to distinguish crawling humans from animals.

The range-weighted target energy is another useful heuristic classification feature. This feature weights the target bulk scatterer (e.g., torso) energy with the square of the target range. The target bulk scatterer energy is

$$E_{bulk}[n] = \left|STFT[k_{bulk}, n]\right|^2 \tag{9.13}$$

where k_{bulk} is the center frequency bin of the bulk-scatterer return. The range-weighted target energy is then

$$E_{range}[n] = E_{bulk}[n]R^2[n] \tag{9.14}$$

where $R[n]$ is the range to the target. The range-weighted target energy feature provides an indirect estimate of a target's size.

9.4 FEATURE SELECTION

Feature selection is the process of pruning features acquired in the feature extraction stage into an efficient set for classification. Due to the inescapable limitations of finite sample size and finite computational resources, feature selection is a critical step in the classification process. The larger the selected feature set becomes, the harder it is for a classifier to accurately model a given class with a fixed training set size. This "curse of dimensionality" effect must be balanced with the need to include enough classification features for high-performance discrimination. Feature selection may be motivated by either the need to reduce the computational burden on the feature extractor or the desire to optimize classification performance.

Stated formally, feature selection is the process of selecting a subset of k features from a superset of n features. The natural goal is to obtain the subset that is *optimal* for classification purposes. If it is assumed that the subset size k is known, the number of subsets of size k from a superset of size n is $\binom{n}{k} = \frac{n!}{k!(n-k)!}$. Feature selection is further complicated by the fact that the optimal subset size k is unknown. As proven by Cover and Van Campenhout [54], exhaustive search is *required* if one desires optimal feature selection with no restriction of set monotonicity (increasing the subset size never increases the probability of error). For this dissertation work, a total of 80 features were extracted from the micro-Doppler data. As an example, choosing 10 features from the 80 feature superset would require training the classifier over $\binom{80}{10} = 1.6 \times 10^{12}$ times. Exhaustive search is clearly computationally prohibitive in this case. The branch and bound algorithm [55], [56] is an optimal feature selection method that significantly reduces computational requirements compared to exhaustive search. However, branch and bound is only optimal under the restriction of set monotonicity. Unfortunately, set monotonicity is typically violated in practice—increasing the number of features presented to a classifier may increase its classification error.

Many sub-optimal sequential search algorithms exist, such as sequential backward selection (SBS), sequential forward selection (SFS), sequential floating backward selection (SFBS), and sequential floating forward selection (SFFS) which perform feature selection in significantly less time than exhaustive search [57], [58], [59], [60], [61]. The sequential search algorithms are based on the "greedy hill-climbing" concept. The "forward" algorithms start with an empty set of features and sequentially add one feature at a time. The single feature added at each step is the one that maximizes the improvement in classification performance. The procedure is repeated until little or no improvement is made. The "backward" algorithms begin with the complete set of extracted features (if computationally feasible) and sequentially remove one feature at a time. The single feature removed at each step is the feature that degrades classification performance the least. This procedure stops when removing any single feature from the set causes a significant (application-defined) reduction in classification performance. The “floating” algorithms simply allow the process to take steps in the reverse direction.

In addition to sequential selection, stochastic search methods such as simulated annealing (SA) [62], [63] and genetic algorithms (GA) [60], [64], [65], [66], [67] have been investigated for feature selection. These approaches are similar in nature to the sequential search methods, but they perform random instead of deterministic selection of features at each stage. Significant computational savings can be obtained by using either the sequential (deterministic) or stochastic search algorithms when compared to exhaustive searching.

This dissertation utilizes a Fisher score initialized sequential backward selection (FSISBS) feature selection algorithm. The Fisher score [63] is defined as the ratio of the between-class scatter matrix $\mathbf{S}_B$ to the average within-class scatter matrix $\mathbf{S}_W$. For the k-class problem, the average within-class scatter matrix is defined as

$$\mathbf{S}_W = \frac{1}{k}\sum_{i=1}^{k}\mathbf{S}_i \quad \text{where} \quad \mathbf{S}_i = \sum_{\mathbf{x}\in D_i}(\mathbf{x}-\mathbf{m}_i)(\mathbf{x}-\mathbf{m}_i)^T \quad \text{and} \quad \mathbf{m}_i = \frac{1}{n_i}\sum_{\mathbf{x}\in D_i}\mathbf{x} \tag{9.15}$$

where n_i is the size of the feature vector set D_i that makes up class i. Likewise, the between-class scatter matrix is defined as

$$\mathbf{S}_B = \sum_{i=1}^{k} n_i(\mathbf{m}_i-\mathbf{m})(\mathbf{m}_i-\mathbf{m})^T \quad \text{and} \quad \mathbf{m} = \frac{1}{n}\sum_{\mathbf{x}}\mathbf{x} \tag{9.16}$$

where n is the total number of feature vectors and $\mathbf{m}$ is the mean vector of the entire feature set. The one-dimensional Fisher score is then

$$\text{Fisher Score (1D)} = \frac{diag(\mathbf{S}_B)}{diag(\mathbf{S}_W)} \tag{9.17}$$

where $diag(\cdot)$ extracts the diagonal elements of a matrix. As shown in [68], the Fisher score is quite limited in its feature selection ability. Features that have little to no classification ability on their own may have excellent classification utility when grouped together. Similarly, features that have good classification properties on their own, but are highly correlated, may interact and perform poorly when grouped together. The one-dimensional Fisher score does not take into account the covariance of features and is thus insufficient as the only means of feature selection. However, the Fisher score can be used as an initialization step for feature selection algorithms. This initialization step is needed when starting a "backward" selection algorithm with the entire feature set is impractical.

The feature selection algorithm begins by calculating the Fisher score of all extracted features. The Fisher score is then thresholded to select the features with the highest one-dimensional discriminatory power. After thresholding the Fisher score, the feature set may contain many useless or redundant features. Sequential backward selection (SBS) is then applied to further refine the feature set. To illustrate the procedure further, each step will be elaborated as it applies to this dissertation. Figure 9.5 shows the Fisher score of the 80 extracted features.

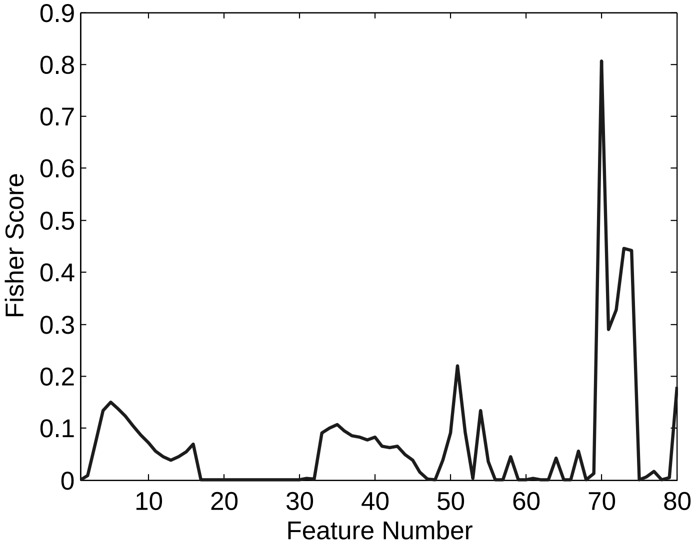


Figure 9.5. Fisher score of all 80 extracted features

Features 1 through 16 are the static cepstral coefficients. Features 17 through 32 are the delta cepstral coefficients, and features 33 through 48 are the delta-delta cepstral coefficients. Features 49 through 57 are the cepstral energy, LPC residual energy, the short-time Fourier transform energy, and their temporal derivatives. Features 58 through 80 are a collection of heuristic features. The peak in the Fisher score is from the range-weighted target energy feature.

The initial feature subset is created by retaining all features with a Fisher score above a threshold of 0.05. This threshold reduces the number of features from 80 down to 33. Since the Fisher score does not account for feature covariance, SBS is used to remove any redundant, highly correlated features. The stopping criterion for the SBS algorithm is to stop removing features when the maximum improvement (relative to the previous step) in F-measure is less than 0.5%. The SBS procedure reduced the number of features from 33 to 12. The final set of FSISBS selected features is shown in Table 9.1.

Table 9.1. FSISBS selected classification features.

Feature Name	Symbol (s)
cepstral coefficient 6, 10, 11, 15, and 16	c_6, c_{10}, c_{11}, c_{15}, and c_{16}
delta-delta cepstral coefficient 11	c''_{11}
delta cepstral energy	$E_{delta-cepstral}$
delta-delta cepstral energy	$E_{delta-delta-cepstral}$
LPC residual energy	$E_{LPC,residual}$
micro-Doppler energy	$E_{micro-Doppler}$
range-weighted target energy	E_{range}
micro-Doppler period	$T_{micro-Doppler}$

9.5 THE CONFUSION MATRIX AND CLASSIFIER PERFORMANCE METRICS

Confusion matrices are often used to evaluate the performance of classifiers. A confusion matrix simply lists the number of examples chosen to be class X when the actual class was class Y. For example, the simple two-class confusion matrix of Table 9.2 shows the results of a hypothetical classification problem.

Table 9.2. Confusion matrix example 1

		Chosen Class	
		Target	Nuisance
Actual Class	Target	9	1
	Nuisance	10	990

An easy way to read confusion matrices is to start by reading the row class label and then read the column class label. The classifier decision is then read, "When the actual class was (row class label), the classifier chose (column class label) N times." For example, from Table 9.2, "When the actual class was Target, the classifier chose Nuisance 1 time." Table 9.3 labels the four possible decisions that are quantified in a two-class confusion matrix.

Table 9.3. Labeled confusion matrix

		Chosen Class	
		Target	Nuisance
Actual Class	Target	*True Positive (TP)*	*False Negative (FN)*
	Nuisance	*False Positive (FP)*	*True Negative (TN)*

Some common performance metrics are

$$\text{specificity} = \frac{TN}{TN + FP} \tag{9.18}$$

$$\text{sensitivity} = \mathrm{P}_{\text{detection}} = \text{recall} = \frac{TP}{TP + FN} \tag{9.19}$$

$$\text{precision} = \frac{TP}{TP + FP} \tag{9.20}$$

where the probability of false alarm (nuisance alarm) is equal to $1 - \text{specificity}$. These measures are typically used in pairs to analyze the performance of binary classifiers. However, classifier training algorithms often require a single metric for comparison of classifier performance. The pattern recognition community often uses accuracy as the single classifier performance measure. Accuracy is defined as

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{9.21}$$

for binary classifiers. One must be careful when analyzing accuracy results (on highly imbalanced datasets) since accuracy is biased toward classes with the largest number of members. $\mathrm{F}_\alpha\text{-measure}$ is an alternative metric that weights the contribution of specificity and sensitivity. For binary classifiers, $\mathrm{F}_\alpha\text{-measure}$ is defined as [69]

$$\mathrm{F}_\alpha\text{-measure} = \frac{(1+\alpha)(\text{specificity} \cdot \text{sensitivity})}{(\alpha \cdot \text{specificity} + \text{sensitivity})} \tag{9.22}$$

where α is a non-negative real constant that weights the contribution of specificity and sensitivity. Note that $\mathrm{F}_0\text{-measure} = \text{specificity}$ and $\mathrm{F}_\infty\text{-measure} = \text{sensitivity}$. The most common $\mathrm{F}_\alpha\text{-measure}$ classification metric is the $\mathrm{F}_1\text{-measure}$ which weights specificity and sensitivity equally. The $\mathrm{F}_1\text{-measure}$ will be referred to as simply the F-measure for the remainder of this dissertation.

Return now to the example of Table 9.2. One calculates that the accuracy is 98.9% and the F-measure is 94.3%. Now consider the example of Table 9.4.

Table 9.4. Confusion matrix example 2

		Chosen Class	
		Target	Nuisance
Actual Class	Target	1	9
	Nuisance	10	990

The accuracy is 98.1% and the F-measure is 18.2%. This example shows that the accuracy metric must be used with caution on highly imbalanced datasets. Confusion matrices and accuracy generalize easily to multi-class classification. Accuracy is calculated as the sum of the main diagonal elements of the confusion matrix divided by the sum of the entire confusion matrix. One can utilize the F-measure in multi-class classification if the problem is broken down into a binary classification task. For the purposes of this dissertation, the human and vehicle classes are considered "classes of interest" that should be detected. The animal class is considered a "nuisance class." The F-measure will score the classifiers' primary task of rejecting the nuisance class.

9.6 POPULATION PARAMETER ESTIMATION USING SAMPLE STATISTICS

In classifier design, one often desires the values of various population parameters (e.g., the population mean and population standard deviation of a Gaussian random variable). However, exact calculation of a population parameter is typically impractical since the entire population must be included in the calculation. For practical reasons, sample statistics (sample mean and sample standard deviation) are used to estimate the value of population parameters.

Let X be a continuous random variable whose domain is the real numbers. The population mean (expected value) is defined as

$$\mu_X = E[X] = \int_{-\infty}^{\infty} x \, f_X(x) \, dx \tag{9.23}$$

where $E[\cdot]$ is the expectation operator and $f_X(x)$ is the probability density function for the random variable X. The population variance is defined as

$$\sigma_X^2 = Var[X] = \int_{-\infty}^{\infty} \left(x - E[X]\right)^2 f_X(x)dx \tag{9.24}$$

where the population standard deviation is σ_X. Obtaining an infinite sample size for the exact calculation of equation (9.23) and equation (9.24) is impractical. Thus, the sample mean and sample standard deviation are used as point estimators of the population mean and population standard deviation, respectively.

The sample mean estimator $\hat{\bar{X}}$ is used to estimate the population mean μ_X. The sample mean estimator is defined as

$$\hat{\bar{X}} = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{9.25}$$

where n is the number of random samples X_i. Before a test is performed, the value of the samples X_i are unknown. Each X_i is a random, independent sample of the population that has the same distribution as the random variable X such that $E[X_i] = E[X] = \mu_X$ and $Var[X_i] = Var[X] = \sigma_X^2$. Thus, the sample mean estimator is itself a random variable with its own mean and variance. The expected value of the sample mean is

$$E[\hat{\bar{X}}] = E\left[\frac{1}{n}\sum_{i=1}^{n} X_i\right] = \frac{1}{n}\sum_{i=1}^{n} E[X_i] = \frac{1}{n} n\mu_X = \mu_X \tag{9.26}$$

where the linearity property of the expectation operator has been used. An estimator is said to be unbiased if the expected value of the estimator is equal to the population parameter. Therefore, the sample mean estimator is an unbiased estimator for the population mean. The variance of the sample mean estimator is

$$\sigma_{\hat{\bar{X}}}^2 = Var\left[\frac{1}{n}\sum_{i=1}^{n} X_i\right] = \frac{1}{n^2} Var\left[\sum_{i=1}^{n} X_i\right] = \frac{1}{n^2}\sum_{i=1}^{n} Var[X_i] = \frac{1}{n^2} n\sigma_X^2 = \frac{\sigma_X^2}{n} \tag{9.27}$$

where the independence of the samples X_i has been assumed.

The sample variance estimator $\hat{S}_X^2$ is used to estimate the population variance σ_X^2. The sample variance estimator is defined as

$$\hat{S}_X^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \hat{\bar{X}})^2 \ . \tag{9.28}$$

The sample variance estimator is an unbiased estimator for the population variance that can be shown (when the X_i are Gaussian) to be distributed as a χ_r^2 (chi-squared) random variable with $r = n-1$ degrees of freedom (independent information units) [70].

Once the experiment has been performed, the samples X_i are no longer random. The observed values for the samples X_i are labeled as x_i. A particular realization for the sample mean estimator $\hat{\bar{X}}$ is the sample mean point estimate $\hat{\mu}_X$ defined as

$$\hat{\mu}_X = \frac{1}{n}\sum_{i=1}^{n} x_i \ . \tag{9.29}$$

Similarly, a realization for the sample variance estimator $\hat{S}_X^2$ is the sample variance point estimate $\hat{\sigma}_X^2$ defined as

$$\hat{\sigma}_X^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \hat{\mu}_X)^2 \ . \tag{9.30}$$

The point estimates $\hat{\mu}_X$ and $\hat{\sigma}_X^2$ are commonly used in the calculation of confidence intervals and statistical significance tests.

9.7 DISTRIBUTION OF THE SAMPLE MEAN ESTIMATOR WHEN σ_X IS KNOWN

The calculations of the previous section utilized the assumption that the random samples X_i were independent and identically distributed. However, no restrictions were made on the type of distribution for the X_i. This section makes the additional assumption that the X_i are Gaussian distributed with population mean μ_X and population standard deviation σ_X. Using shorthand notation, the X_i are distributed as

$$X_i \sim N(\mu_X, \sigma_X). \tag{9.31}$$

The distribution of the sample mean estimator $\hat{\bar{X}}$ can be found using the Gaussian assumption for the X_i. If σ_X is known, then $\hat{\bar{X}}$ is a Gaussian distributed random variable since $\hat{\bar{X}}$ is the sum of n Gaussian random variables. The distribution of $\hat{\bar{X}}$ is

$$\hat{\bar{X}} \sim N\left(E[\hat{\bar{X}}], \sigma_{\hat{\bar{X}}}\right) \quad \sim N\left(\mu_X, \frac{\sigma_X}{\sqrt{n}}\right) \qquad (\sigma_X \text{ known}) \tag{9.32}$$

where the last step used equations (9.26) and (9.27). The random variable Z is commonly defined as the normalized (zero mean, unit standard deviation) version of $\hat{\bar{X}}$ such that

$$Z = \frac{\hat{\bar{X}} - E[\hat{\bar{X}}]}{\sigma_{\hat{\bar{X}}}} = \frac{\hat{\bar{X}} - \mu_X}{\frac{\sigma_X}{\sqrt{n}}} \quad \Rightarrow \quad Z \sim N(0,1) \qquad (\sigma_X \text{ known}). \tag{9.33}$$

The random variable Z is said to have a standard normal (Gaussian) distribution. The point estimate for the random variable Z is the z-score test statistic defined as

$$z = \frac{\hat{\mu}_X - \mu_X}{\frac{\sigma_X}{\sqrt{n}}} \qquad (\sigma_X \text{ known}) . \tag{9.34}$$

The z-score is commonly used to estimate confidence intervals around the population mean μ_X and to perform statistical significance testing (comparing the sample mean $\hat{\mu}_X$ to the population mean μ_X).

9.8 DISTRIBUTION OF THE SAMPLE MEAN ESTIMATOR WHEN σ_X IS UNKNOWN

Unfortunately, the population standard deviation is rarely known. Equation (9.34) is applicable in practice only when the sample size is large ($n \geq 30$ by practical convention). As the sample size gets larger, a sample estimate approaches the true value of the population parameter.

When σ_X is unknown, the distribution of the sample mean estimator is the Student's t-distribution. The random variable T_r uses the sample standard deviation estimator $\hat{S}_X$ in place of σ_X. The random variable T_r is defined as

$$T_r = \frac{\hat{\bar{X}} - \mu_X}{\frac{\hat{S}_X}{\sqrt{n}}} \qquad r = n - 1 \tag{9.35}$$

where r is the number of degrees of freedom. The number of degrees of freedom (which quantify the amount of independent information) is only $n-1$ since one degree of freedom is lost by using the sample mean estimator to calculate the sample standard deviation estimator. The Student's t-distribution for the random variable T_r is [70]

$$f_{T_r}(t,r) = \frac{1}{\sqrt{\pi r}} \frac{\Gamma\left(\left(r+1\right)/2\right)}{\Gamma\left(r/2\right)} \left(1+\frac{t^2}{r}\right)^{-[(r+1)/2]} \qquad -\infty < t < \infty, \quad r > 0 \tag{9.36}$$

where $\Gamma(\cdot)$ is the gamma function. Figure 9.6 shows the Student's t-distribution with 9 degrees of freedom.

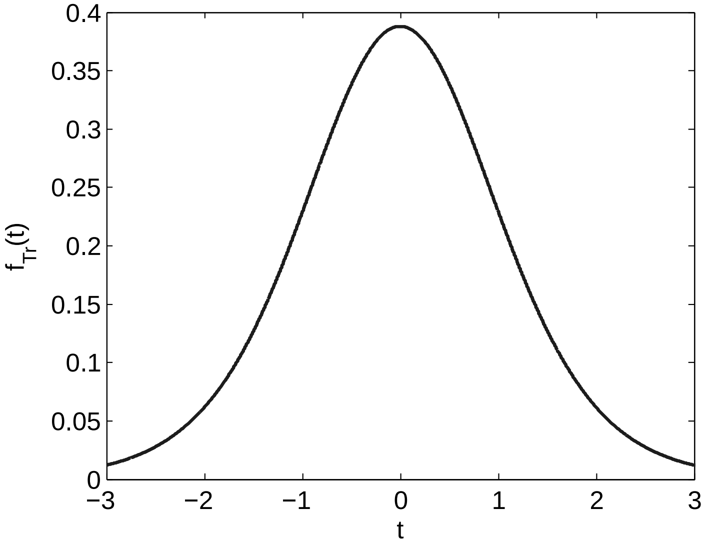


Figure 9.6. Student's t-distribution with 9 degrees of freedom

The Student's t-distribution is similar to the standard normal distribution except that the tails of the t-distribution are higher. The point estimate for T_r is the t_r test statistic defined as

$$t_r = \frac{\hat{\mu}_X - \mu_X}{\frac{\hat{\sigma}_X}{\sqrt{n}}} \qquad r = n-1. \tag{9.37}$$

The t_r test statistic is useful for estimating confidence intervals and evaluating statistical significance when the population standard deviation is unknown.

9.9 POPULATION MEAN CONFIDENCE INTERVAL ESTIMATORS

Although finding the exact value of the population mean is impractical, one often estimates an interval which contains the population mean with a specified probability (confidence level). Note that care must be taken when evaluating the meaning of a confidence interval estimate. Only the confidence interval *estimator* (which is a *random interval*) contains the population mean at the specified confidence level probability. Once the experiment has been performed, the confidence interval estimate is no longer random and interpreting the results in terms of probabilities must be performed with caution (or avoided).

To assist in confidence interval calculation, the value $t_{r,\alpha/2}$ is defined such that

$$P\left[-t_{r,\alpha/2} < T_r < t_{r,\alpha/2}\right] = F_{T_r}\left(t_{r,\alpha/2}\right) - F_{T_r}\left(-t_{r,\alpha/2}\right) = \int_{-t_{r,\alpha/2}}^{t_{r,\alpha/2}} f_{T_r}(t,r)\,dt = 1-\alpha \qquad (9.38)$$

where $F_{T_r}(\cdot)$ the is the cumulative distribution function (CDF) of the random variable T_r. Integrals such as the one in equation (9.38) can be evaluated using either numerical simulation software or a standard t-table. Equation (9.38) can be used to derive the confidence interval estimator as follows

$$\begin{aligned}
&P\left[-t_{r,\alpha/2} < T_r < t_{r,\alpha/2}\right] = 1-\alpha \\
&P\left[-t_{r,\alpha/2} < \frac{\hat{\bar{X}} - \mu_X}{\hat{S}_X \big/ \sqrt{n}} < t_{r,\alpha/2}\right] = 1-\alpha \\
&P\left[-t_{r,\alpha/2}\frac{\hat{S}_X}{\sqrt{n}} < \hat{\bar{X}} - \mu_X < t_{r,\alpha/2}\frac{\hat{S}_X}{\sqrt{n}}\right] = 1-\alpha \\
&P\left[\hat{\bar{X}} - t_{r,\alpha/2}\frac{\hat{S}_X}{\sqrt{n}} < \mu_X < \hat{\bar{X}} + t_{r,\alpha/2}\frac{\hat{S}_X}{\sqrt{n}}\right] = 1-\alpha
\end{aligned} \qquad (9.39)$$

where the last equation is the confidence interval estimator for a $1-\alpha$ confidence level. Due to the inclusion of the random variables $\hat{\bar{X}}$ and $\hat{S}_X$, the confidence interval estimator is a random interval. After an experiment has been performed, the observed values of the sample mean and sample standard deviation are not random. The sample estimates are used to calculate the following $1-\alpha$ confidence interval estimate

$$\hat{\mu}_X - t_{r,\alpha/2}\frac{\hat{\sigma}_X}{\sqrt{n}} < \mu_X < \hat{\mu}_X + t_{r,\alpha/2}\frac{\hat{\sigma}_X}{\sqrt{n}} \,. \tag{9.40}$$

This dissertation will make use of the confidence interval and statistical significance testing (discussed in the next section) to evaluate the performance of one classifier relative to another.

9.10 STATISTICAL SIGNIFICANCE TESTING FOR CLASSIFIER COMPARISON

In this dissertation, statistical significance testing (along with confidence intervals) is utilized to facilitate comparison of the support vector machine (SVM) and Gaussian mixture model (GMM) classifiers. This section discusses the theory underlying the statistical significance tests used in this dissertation. A common question that arises during the design of a pattern recognition system is whether two classification algorithms significantly differ from one another on a given classification metric. When comparing classifiers, it must be explicitly stated what is meant by the term “significant” because many (often conflicting) definitions exist. This dissertation will follow common practice in the pattern recognition community and analyze whether the difference in classification accuracy between two algorithms is “statistically significant.” Statistical significance tests analyze whether differing experimental outcomes are likely to be a result of random influences only. Statistical significance tests should be interpreted with caution. There is no universally accepted method to interpret statistical significance tests, and even experts in the field disagree on proper interpretation. However, statistical significance testing is included in this dissertation because of its ubiquitous use in the literature. The reader is also cautioned to make a clear mental distinction between the two theories of Neyman-Pearson hypothesis testing (as discussed in Section 5.2) and statistical significance testing (the subject of this section). Neyman-Pearson hypothesis testing is performed to optimize the decision making process (e.g., maximize detection probability at a fixed false alarm probability). Significance testing is a process of statistical inference (inferring relationships among population parameters from their sample estimates). A detailed discussion of the differences between Neyman-Pearson hypothesis testing and statistical significance testing can be found in [71] and [72].

Assume that the accuracy of two classification algorithms (SVM and GMM) is to be compared. The goal of significance testing is to determine if there is a statistically significant difference between the (unknown) *population* mean accuracy of each algorithm. Note that metrics other than classification accuracy can also be used. The null hypothesis is that the population means are equal. Stated formally, the null hypothesis is

$$H_0 : \mu_{X,SVM} = \mu_{X,GMM} \qquad \Rightarrow \qquad H_0 : \mu_{X,SVM} - \mu_{X,GMM} = \mu_D = 0 . \tag{9.41}$$

where $\mu_{X,SVM}$ is the population mean accuracy of the SVM classifier, $\mu_{X,GMM}$ is the population mean accuracy of the GMM classifier, and μ_D is the difference in population mean accuracy. Note that unlike in Neyman-Pearson hypothesis testing, the alternative hypothesis is not specified for significance testing. (The lack of an alternative hypothesis in significance testing was the chief complaint of the proponents of hypothesis testing

[71].) The null hypothesis will either be accepted (no statistically significant difference) or rejected (the difference is statistically significant) based on the statistical test.

The first step in statistical significance testing is to perform n test runs on disjoint data subsets (see Chapter 12). If both algorithms are applied to the same n data subsets, a *paired* significance test is performed. This dissertation utilizes a paired Student's t-test for statistical significance testing. In a paired statistical test, D is a random variable that represents the difference in accuracy between the two algorithms on the same data subset such that

$$D = X_{SVM} - X_{GMM} \tag{9.42}$$

where X_{SVM} and X_{GMM} are random variables representing the accuracy of the SVM and GMM classifiers, respectively. The t_r test statistic is calculated as

$$t_r = \frac{\hat{\mu}_D - \mu_D}{\frac{\hat{\sigma}_D}{\sqrt{n}}} \qquad r = n-1 \tag{9.43}$$

where $\hat{\mu}_D$ and $\hat{\sigma}_D$ are point estimates for the sample mean and sample standard deviation calculated from the *difference* in classifier accuracy. The null hypothesis assumption ($\mu_D = 0$) allows the t_r test statistic to be calculated using only point estimates.

The next step in the significance test is to calculate the $p-\text{value}$ for the paired Student's t-test. The $p-\text{value}$ is the probability that the sample mean estimators, $\hat{\bar{X}}_{SVM}$ and $\hat{\bar{X}}_{GMM}$, differ by at least as much as the observed difference, $\hat{\mu}_{X,SVM} - \hat{\mu}_{X,GMM}$ (under the assumption that the null hypothesis H_0 is true). In equation form, the $p-\text{value}$ is defined as

$$
\begin{aligned}
p &= P\left(\left|\hat{\bar{X}}_{SVM} - \hat{\bar{X}}_{GMM}\right| \geq \hat{\mu}_{X,SVM} - \hat{\mu}_{X,GMM} \left| H_0 \right.\right) \\
&= P\left(\left|\hat{\bar{D}}\right| \geq \hat{\mu}_D \left| H_0 \right.\right) \\
&= P\left(\left|T_r\right| \geq t_r \left| H_0 \right.\right) \qquad (9.44) \\
&= 1 - P\left(t_r \leq T_r \leq t_r \left| H_0 \right.\right) \\
&= 1 - \int_{-t_r}^{t_r} f_{T_r}(t,r)\,dt\,.
\end{aligned}
$$

The area of the shaded regions in Figure 9.7 represents the $p-$value (for $t_9 = 1.5$).

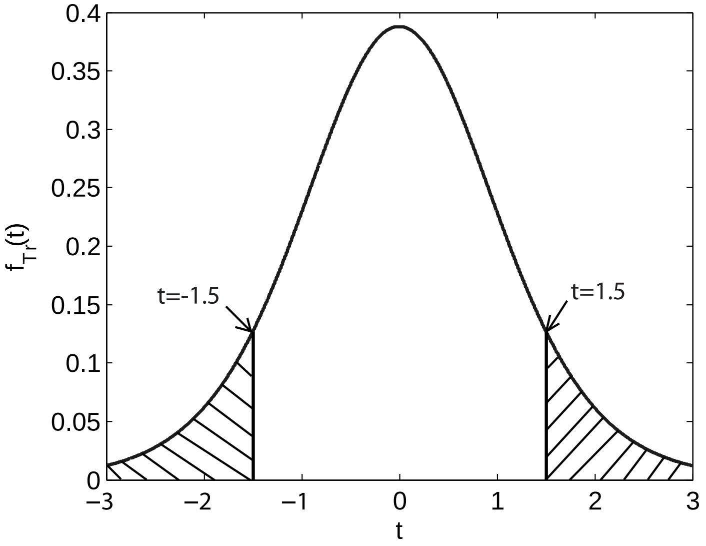


Figure 9.7. The Student's t-test ($t_9 = 1.5$) $p-$value is the area of the shaded region.

The $p-$value provides evidence against the null hypothesis. The smaller the $p-$value becomes, the less likely it is that the null hypothesis is true. The null hypothesis is rejected and the difference in performance is labeled "statistically significant" if the

$p-$value is below a specified threshold. Although there is much contention over the interpretation of the $p-$value, the following thresholds are quite common

$$\begin{aligned} &p > 0.05 && \text{"not statistically significant"} \\ &p < 0.05 && \text{"statistically significant"} \\ &p < 0.01 && \text{"highly statistically significant".} \end{aligned} \tag{9.45}$$

Note that even if significance testing results in high confidence of *statistical* significance, more analysis is required to determine if the results of the experiment are *practically* significant. Statistical significance only suggests that an underlying phenomenon other than random influence is likely to exist. However, statistical significance does not answer the important question of *what* the underlying phenomenon is.

Chapter 12 presents both the $p-$value and various confidence intervals for classifier performance comparison. Chapter 10 and Chapter 11 present the theory of support vector machine (SVM) and Gaussian mixture model classification, respectively.

Chapter 10: Support Vector Machines

10.1 INTRODUCTION TO SUPPORT VECTOR MACHINES

This section will introduce the basic ideas behind support vector machines (SVMs) as developed by V. Vapnik, et al. The approach will follow the developments in the classical SVM work [73] and the excellent tutorial in [74]. Support vector machines (SVMs) are quite general learning machines introduced to solve problems in the fields of pattern recognition, regression estimation, and density estimation. The development presented here will focus on the pattern recognition problem.

Figure 10.1 illustrates the basic concepts of support vector machines.

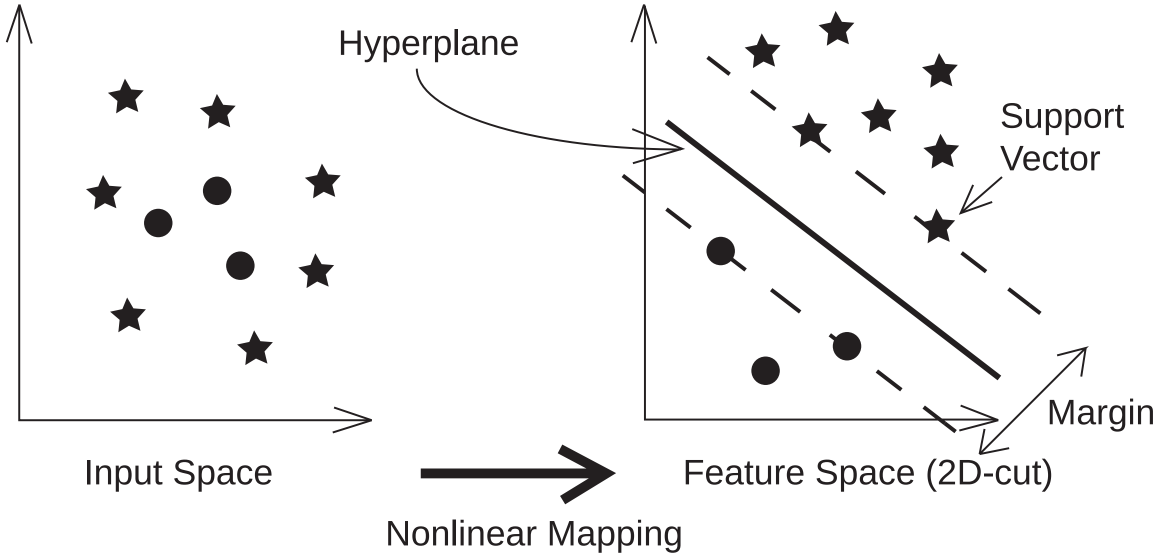


Figure 10.1. Support vector machines

Support vector machines take a set of input features that are typically not linearly separable and transform the features into a higher dimensional feature space. In this new space, the data are separated by a linear hyperplane. (As discussed later in Section 10.2, the complexity of the resulting algorithm will depend on the number of support vectors and not the dimensionality of the feature space.) In Figure 10.1, the support vectors are the feature vectors lying on the two dashed hyperplanes. The support vectors provide the most information for the classification task but they are also the feature vectors that are hardest to classify. Many hyperplanes can separate the data in Figure 10.1. However, the goal of support vector machines is to find the *optimal* hyperplane (largest margin hyperplane) so that the classifier will perform well on both the training set and unseen test samples.

Support vector machines are a supervised learning technique (learning from examples). The supervisor (SVM designer) provides a set of l labeled training data vectors (each of dimension d)

$$(\mathbf{x}_i, y_i) \quad i = 1, \cdots, l \quad \mathbf{x}_i \in \mathbb{R}^d \quad y_i \in \{-1, 1\} \tag{10.1}$$

where the $\mathbf{x}_i$ are the training data vectors and the y_i are the class labels assigned by a supervisor (note the use of bold font for vectors). The general goal of SVMs is to find the optimal hyperplane (decision rule) that separates the training data well and also generalizes appropriately for unseen test data.

Figure 10.2 gives a pictorial view of the important SVM parameters.

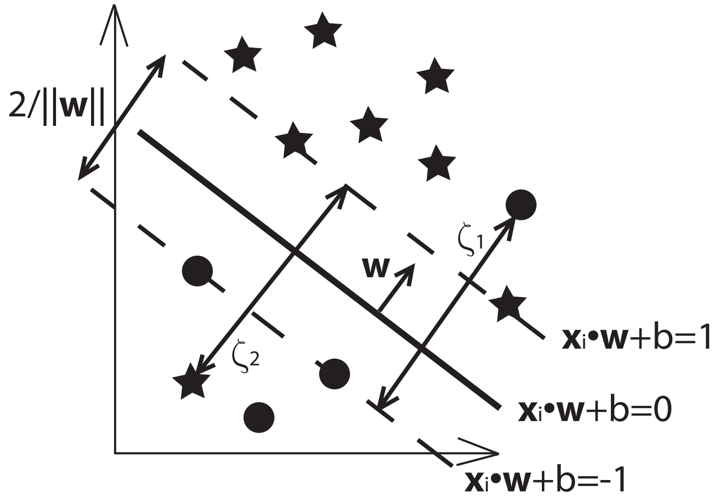


Figure 10.2. Adding slack variables to avoid over-fitting training data

The support vectors lie on planes defined by the equations $y_i\left(\mathbf{x}_i \cdot \mathbf{w} + b\right) = 1$. Thus, the "hard" margin is equal to $\frac{2}{\|\mathbf{w}\|}$. Support vector machines provide the optimal separating hyperplane by maximizing the margin. In the case of separable data, the hard margin is maximized which involves minimizing $\|\mathbf{w}\|$ (that is, minimizing the norm of the hyperplane's normal vector). For non-separable data, the concept of a "soft" margin is introduced [73]. A soft margin is created by adding the non-negative slack variables ζ_i. The slack variables represent the shortest distance between an incorrectly classified training vector and its correct classification region $\zeta_i = \left|y_i - \left(\mathbf{x}_i \cdot \mathbf{w} + b\right)\right|^2$. Slack variables help the SVM to minimize training set errors.

Maximizing the soft margin requires minimizing the following objective function

$$E = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{l}\zeta_i \tag{10.2}$$

subject to the following constraints

$$\mathbf{x}_i \cdot \mathbf{w} + b \geq 1 - \zeta_i \quad \text{for } y_i = 1 \tag{10.3}$$

$$\mathbf{x}_i \cdot \mathbf{w} + b \leq \zeta_i - 1 \quad \text{for } y_i = -1 \tag{10.4}$$

$$\zeta_i \geq 0 \quad \forall i \,. \tag{10.5}$$

The first term on the right hand side of equation (10.2) is the margin maximizing term responsible for the well-known generalization capability of SVMs. The second term on the right hand side of equation (10.2) is related to the empirical risk. Minimizing the empirical risk is tantamount to minimizing the training set error. The number of errors on the training set can be brought down to zero by utilizing a highly complex classifier. However, overly complex classifiers rarely perform well on unseen test samples. The cost parameter C controls the number of support vectors used to model the decision boundary. The greater the number of support vectors, the more complex the decision boundary. As the number of support vectors gets smaller, the decision boundary becomes progressively smoother. A balance must be achieved between using not enough support vectors to model the inherent complexity of the problem, and using too many support vectors which will over-train the model and perform poorly on unseen test data. Obtaining the optimal model complexity is the subject of structural risk minimization as developed in [73].

Returning now to the optimization problem, constraint equations (10.3) and (10.4) can be combined into a single equation

$$y_i \left(\mathbf{x}_i \cdot \mathbf{w} + b \right) \geq 1 - \zeta_i \,. \tag{10.6}$$

The method of Lagrange multipliers will be used to perform this inequality constrained optimization problem. The primal Lagrangian formed from objective function (10.2) and constraint equations (10.5) and (10.6) is

$$L_P = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{l}\zeta_i - \sum_{i=1}^{l}\alpha_i\left[y_i\left(\mathbf{x}_i \cdot \mathbf{w} + b\right) - 1 + \zeta_i\right] - \sum_{i=1}^{l}\beta_i\zeta_i \tag{10.7}$$

where α_i and β_i are the non-negative Lagrange multipliers. The primal Lagrangian is now minimized with respect to $\mathbf{w}$. To find the minimizer $\mathbf{w}^*$, start by solving the following set of simultaneous equations

$$\frac{\partial L_P}{\partial \mathbf{w}} = 0 \quad \frac{\partial L_P}{\partial b} = 0 \quad \frac{\partial L_P}{\partial \zeta_i} = 0 . \tag{10.8}$$

The solution is as follows

$$\begin{aligned} &\frac{\partial L_P}{\partial \mathbf{w}} = \mathbf{w}^* - \sum_{i=1}^{l}\alpha_i y_i \mathbf{x}_i \quad \Rightarrow \quad \mathbf{w}^* = \sum_{i=1}^{l}\alpha_i y_i \mathbf{x}_i \\ &\frac{\partial L_P}{\partial b} = -\sum_{i=1}^{l}\alpha_i y_i \quad \Rightarrow \quad \sum_{i=1}^{l}\alpha_i y_i = 0 \\ &\frac{\partial L_P}{\partial \zeta_i} = C - \alpha_i - \beta_i \quad \Rightarrow \quad C = \alpha_i + \beta_i . \end{aligned} \tag{10.9}$$

The last equation constrains the α_i to a hypercube $0 \le \alpha_i \le C$. The dual Lagrangian can now be formed by substituting equations (10.9) back into the primal Lagrangian. Solving for the dual Lagrangian produces

$$\begin{aligned}L_D &= \frac{1}{2}\left(\sum_{i=1}^{l}\alpha_i y_i \mathbf{x}_i\right)\cdot\left(\sum_{j=1}^{l}\alpha_j y_j \mathbf{x}_j\right) + C\sum_{i=1}^{l}\zeta_i \\ &\quad -\sum_{i=1}^{l}\alpha_i\left[y_i\left(\mathbf{x}_i\cdot\left(\sum_{j=1}^{l}\alpha_j y_j \mathbf{x}_j\right)+b\right)-1+\zeta_i\right]-\sum_{i=1}^{l}\beta_i\zeta_i \\ &= \frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j y_i y_j\left(\mathbf{x}_i\cdot\mathbf{x}_j\right)+C\sum_{i=1}^{l}\zeta_i \\ &\quad -\sum_{i,j=1}^{l}\alpha_i\alpha_j y_i y_j\left(\mathbf{x}_i\cdot\mathbf{x}_j\right)-b\sum_{i=1}^{l}\alpha_i y_i+\sum_{i=1}^{l}\alpha_i-\sum_{i=1}^{l}\alpha_i\zeta_i-\sum_{i=1}^{l}\beta_i\zeta_i\end{aligned} \quad . \qquad (10.10)$$

The following dual Lagrangian and its associated constraints are found by substitution of equations (10.9).

$$\begin{aligned}&L_D = \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j y_i y_j\left(\mathbf{x}_i\cdot\mathbf{x}_j\right) \\ &\sum_{i=1}^{l}\alpha_i y_i = 0 \\ &0\le\alpha_i\le C \quad \forall i\,.\end{aligned} \qquad (10.11)$$

Note that (by a clever choice) the slack variables ζ_i do not show up in the dual Lagrangian. To minimize the objective function of equation (10.2), the dual Lagrangian must be maximized with respect to the α_i. This step is typically performed using either a numerical quadratic programming library or the popular Sequential Minimal Optimization (SMO) algorithm [75]. The minimizer $\mathbf{w}^*$ and the optimal b^* are found from the optimal Lagrange multipliers α_i^*

$$\mathbf{w}^* = \sum_{i=1}^{l} \alpha_i^* y_i \mathbf{x}_i$$
$$b^* = y_k \left(1 - \zeta_k\right) - \sum_{i=1}^{l} \alpha_i^* y_i (\mathbf{x}_i \cdot \mathbf{x}_k) . \tag{10.12}$$

The resulting decision function to be applied to test data is

$$f(\mathbf{x}; \alpha_i^*, b^*) = \sum_{i=1}^{l} \alpha_i^* y_i \left(\mathbf{x}_i \cdot \mathbf{x}\right) + b^*$$
$$\hat{y} = \operatorname{sgn}\left[f(\mathbf{x}; \alpha_i^*, b^*) \right] = \operatorname{sgn}\left[\sum_{i=1}^{l} \alpha_i^* y_i \left(\mathbf{x}_i \cdot \mathbf{x}\right) + b^* \right] \tag{10.13}$$

where $\hat{y} \in \{-1, 1\}$ is the estimated class label and sgn(·) produces the sign of the argument. Note that only the support vectors (those vectors with non-zero α_i^*) contribute to the sum, which greatly reduces the complexity of SVM classifiers.

10.2 NON-LINEAR SUPPORT VECTOR MACHINES FOR NON-SEPARABLE TRAINING DATA

So far, only linear decision functions have been considered. The key idea of support vector machines is to transform the *input space* (which is likely non-separable) into a higher dimensional *feature space* where the training data are linearly separable. The linear decision functions in the feature space are typically non-linear in the input space.

Define a mapping Φ which maps the d-dimensional input space into a higher-dimensional (potentially infinite-dimensional) feature space H.

$$\Phi : \mathbb{R}^d \rightarrow H . \tag{10.14}$$

With this mapping, the inner products $\mathbf{x}_i \cdot \mathbf{x}_j$ in input space become $\Phi(\mathbf{x}_i)\cdot\Phi(\mathbf{x}_j)$ in the feature space. Defining the precise form for Φ is often difficult and/or costly. Thus, we seek a kernel function $K(\mathbf{x}_i,\mathbf{x}_j)=\Phi(\mathbf{x}_i)\cdot\Phi(\mathbf{x}_j)$ so that defining Φ is unnecessary. But how do we know if a particular kernel function exists in feature space? Mercer's Theorem provides us the answer. Mercer's Theorem states that the mapping Φ and the kernel function $K(\mathbf{x},\mathbf{y})=\sum_i \Phi_i(\mathbf{x})\Phi_i(\mathbf{y})$ exist if and only if [74]

$$\begin{gathered}\forall g(\mathbf{x}) \text{ with } \int g(\mathbf{x})^2\, d\mathbf{x} < \infty \text{ then} \\ \int K(\mathbf{x},\mathbf{y})\, g(\mathbf{x}) g(\mathbf{y})\, d\mathbf{x} d\mathbf{y} \geq 0 .\end{gathered} \tag{10.15}$$

Therefore, for all pairs of mappings and kernels that satisfy Mercer's Theorem, the kernel function $K(\mathbf{x}_i,\mathbf{x}_j)$ takes the place of $\mathbf{x}_i \cdot \mathbf{x}_j$ in the non-linear optimization problem. Thus, the results from linear SVMs can immediately be extended to the non-linear case as follows

$$\begin{gathered} f(\mathbf{x};\alpha_i^*,b^*) = \sum_{i=1}^{l} \alpha_i^* y_i K(\mathbf{x},\mathbf{x}_i) + b^* \\ \hat{y} = \operatorname{sgn}\left[f(\mathbf{x};\alpha_i^*,b^*)\right] = \operatorname{sgn}\left[\sum_{i=1}^{l} \alpha_i^* y_i K(\mathbf{x},\mathbf{x}_i) + b^*\right] \end{gathered} \tag{10.16}$$

where the optimal bias b^* is

$$b^* = y_k(1-\zeta_k) - \sum_{i=1}^{l} \alpha_i^* y_i K(\mathbf{x}_i,\mathbf{x}_k) \ . \tag{10.17}$$

Again, note that only the support vectors contribute to the sums. The support vector approach allows us to perform the inner products in high-dimensional feature space using a kernel with complexity determined by the number of support vectors.

There is no known method for selecting the most appropriate kernel function for a given classification task. SVM practitioners typically begin by trying kernels that have been known to produce good results. Some of the most common kernel functions are

$$\text{Polynomial:} \quad K\left(\mathbf{x},\mathbf{x}_i\right)=\left[\gamma(\mathbf{x}\cdot\mathbf{x}_i)+\delta\right]^d \tag{10.18}$$

$$\text{Radial Basis Function (RBF)}: \; K\left(\mathbf{x},\mathbf{x}_i\right)=\exp\left[-\gamma\left\|\mathbf{x}-\mathbf{x}_i\right\|^2\right] \tag{10.19}$$

$$\text{Sigmoid:} \quad K\left(\mathbf{x},\mathbf{x}_i\right)=\tanh\left[\gamma\left(\mathbf{x}\cdot\mathbf{x}_i\right)-\delta\right] \tag{10.20}$$

The kernel function parameters γ, d, and δ are typically chosen by heuristic methods (often by grid-search).

Although SVMs were presented here for solving pattern recognition problems, SVMs have shown promise in solving the more general problems of regression estimation and density estimation. Readers who wish to further investigate the theory of support vector machines should consult the classic work [73] and the comprehensive tutorial in [74].

10.3 PRACTICAL CONSIDERATIONS FOR USING SUPPORT VECTOR MACHINES

Support vector machines have been applied in many practical applications, such as text categorization [76], face detection [77], and genetic research [78]. The SVM classifiers for text categorization are among the top performing algorithms for this task. SVMs achieve good generalization largely due to the construction of an optimal soft-margin hyperplane. When optimizing the SVM in the training phase, the margin-maximization process helps prevent over-fitting the classifier to the training data. SVMs directly incorporate methods of regulating model complexity. Another positive aspect of SVMs is the limited use of heuristics. Many classification algorithms such as neural networks require extensive fine-tuning of a large number of heuristic parameters. The use of heuristics in SVMs is typically limited to finding an appropriate cost parameter and defining the kernel function (for example, finding the γ parameter of the radial basis function kernel).

SVMs exhibit some negative attributes that may preclude them from use in some applications. The training time for support vector machines is typically long compared to many traditional classifiers. The training time might be prohibitive for systems which utilize tens to hundreds of thousands of training samples. The sequential minimal optimization (SMO) algorithm described in [75] was designed to improve SVM training time. Training time is not expected to be of much concern for this dissertation research since the training set size is not excessively large. Execution speed on test data is more important for many applications. SVM execution speed may prevent its use in some applications unless some complexity reduction is performed. The reduced set method was developed in [79] to improve test-phase execution speed by using a reduced set of support vectors in the decision function.

The theoretical development of support vector machines concentrates on binary classification problems. Several methods have been developed to extend SVMs for multi-class problems. One approach to multi-class SVM reformulates the SVM optimization problem to solve the multi-class problem in one step [80]. However, this approach is seldom used in practice because the resulting optimization problem is significantly more complex than the binary optimization problem. Another approach to multi-class SVM is to utilize multiple binary SVMs. One such method (the method used in this dissertation) is the "one-versus-rest" method [81]. In the one-versus-rest method, one SVM is trained for each of the c classes. A distinct SVM is trained for the human, vehicle, and animal classes in this dissertation. As an example, the "human-versus-rest"

SVM is trained by using the human examples as "positive labels" and the vehicle and animal classes as "negative labels." The "vehicle-versus-rest" and "animal-versus-rest" SVMs are trained in a similar manner. To classify a test vector $\mathbf{x}$, each of the c SVMs are applied to $\mathbf{x}$. The test vector $\mathbf{x}$ is assigned the label of the class with the maximum decision function value such that

$$\hat{y} = \arg\max_{j} \left[f(\mathbf{x}; \alpha_{i,j}{}^{*}, b_j^{*}) \right] \qquad j = 1, 2, \cdots, c \tag{10.21}$$

where $\alpha_{i,j}{}^{*}$ and b_j^{*} are the Lagrange multipliers and bias for class j.

Chapter 11: Gaussian Mixture Models

11.1 SEQUENCE CLASSIFICATION: THE MAP AND ML DECISION RULES

Sequence classification estimates which class $\hat{\lambda} \in \{\lambda_1, \lambda_2, \cdots \lambda_c\}$ produced the observed sequence of feature vectors

$$O = \{\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_T\} \tag{11.1}$$

where O is the observation sequence, c is the number of classes, and T is the number of feature vectors per sequence. The symbol λ represents both the class label and the model parameters, but the meaning should be clear from the context. If equal weights are assigned to all types of classification errors, the optimal solution in terms of minimizing the probability of error is the maximum a posteriori (MAP) decision rule. The MAP decision rule is

$$\hat{\lambda}_{MAP} = \arg\max_{\lambda_i} P(\lambda_i \mid O) \quad i = 1, 2, \cdots c \tag{11.2}$$

or equivalently,

$$choose\ \hat{\lambda}_{MAP} = \lambda_i \quad if \quad P(\lambda_i \mid O) > P(\lambda_j \mid O) \quad \forall i \neq j \quad i, j = 1, 2, \cdots c. \tag{11.3}$$

The posterior probability $P(\lambda_i \mid O)$ is the probability that the class λ_i produced the observed sequence O. Maximizing the posterior probability is equivalent to minimizing the Bayes risk with the condition that each type of error is assigned the same risk [63].

In order to solve for the MAP solution, the posterior probability $P(\lambda_i \mid O)$ must be calculated. From the definition of conditional probability, we have

$$P(\lambda_i \mid O) = \frac{p(\lambda_i, O)}{p(O)} \tag{11.4}$$

and

$$p(O \mid \lambda_i) = \frac{p(\lambda_i, O)}{P(\lambda_i)} \tag{11.5}$$

where capital letters refer to probability mass functions, lowercase letters refer to probability density functions, and $p(\lambda_i, O)$ is the probability that both the sequence O and the class λ_i are simultaneously observed. Substituting equation (11.5) into the right hand side of equation (11.4) results in Bayes theorem for sequence classification

$$P(\lambda_i \mid O) = \frac{p(O \mid \lambda_i) P(\lambda_i)}{p(O)}. \tag{11.6}$$

The prior probability $P(\lambda_i)$ is the probability of the class λ_i being the correct class before any experiment is undertaken. The likelihood $p(O \mid \lambda_i)$ is the probability that the sequence O is observed, given that the class λ_i produced O. The total probability $p(O)$ is simply a normalization factor to ensure that the posterior distribution is a valid probability mass function. That is, $p(O)$ enforces the following stochastic constraint

$$\sum_{k=1}^{c} P(\lambda_k \mid O) = 1. \tag{11.7}$$

The probability $p(O)$ can be expanded by employing the law of total probability

$$p(O) = \sum_{k=1}^{c} p(\lambda_k, O) = \sum_{k=1}^{c} p(O \mid \lambda_k) P(\lambda_k). \tag{11.8}$$

Since $p(O)$ is equal for all classes, it cancels out in the decision rule. By substitution of equation (11.6), the MAP decision rule becomes

$$\hat{\lambda}_{MAP} = \arg\max_{\lambda_i} p(O \mid \lambda_i) P(\lambda_i) \quad i = 1, 2, \cdots c \tag{11.9}$$

or equivalently,

$$choose\ \hat{\lambda}_{MAP} = \lambda_i \quad if \quad p(O \mid \lambda_i) P(\lambda_i) > p(O \mid \lambda_j) P(\lambda_j) \quad \forall i \neq j \quad i, j = 1, 2, \cdots c\,. \tag{11.10}$$

In the case that all classes have equal prior probabilities $P(\lambda_i)$, the MAP decision rule simplifies to the maximum likelihood (ML) decision rule. The ML decision rule is

$$\hat{\lambda}_{ML} = \arg\max_{\lambda_i} p(O \mid \lambda_i) \quad i = 1, 2, \cdots c \tag{11.11}$$

or equivalently,

$$choose\ \hat{\lambda}_{ML} = \lambda_i \quad if \quad p(O \mid \lambda_i) > p(O \mid \lambda_j) \quad \forall i \neq j \quad i, j = 1, 2, \cdots c\,. \tag{11.12}$$

A designer may choose to implement the ML decision rule if estimating the prior probabilities is too costly or impossible. In addition, the ML decision rule may be favored over the MAP decision rule if the classifier is to operate under widely varying operational conditions. Class priors may be estimated from the training data, but the actual class prior distribution in operational conditions may be drastically different than what is represented in the training set. For this reason, the generative classifiers in this dissertation will utilize the maximum likelihood decision rule.

The MAP and ML decision rules can be further simplified if we assume that each observed feature vector is statistically independent. The assumption of statistically independent feature vectors implies that

$$p(\vec{x}_{t_1}, \vec{x}_{t_2} \mid \lambda_i) = p(\vec{x}_{t_1} \mid \lambda_i) p(\vec{x}_{t_2} \mid \lambda_i) \quad t_1 \neq t_2\,. \tag{11.13}$$

This assumption states that each *sequential* observation in time is independent of any other observation. Although equation (11.13) is often violated, the computational savings is typically significant enough in practice to warrant ignoring the correlation among feature vectors observed at different times. Note that the assumption of statistically independent feature vectors is *not* the same as assuming that each of the d elements of a d-dimensional feature vector is independent of each other (the standard assumption of naïve Bayes classifiers). The sequence likelihood function $p(O\,|\,\lambda_i)$ (also known as the likelihood score) factors into the product of individual feature vector likelihood functions $p(\vec{x}_t\,|\,\lambda_i)$ such that

$$p(O\,|\,\lambda_i) = p(\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_T\,|\,\lambda_i) = \prod_{t=1}^{T} p(\vec{x}_t\,|\,\lambda_i) \ . \tag{11.14}$$

The likelihood function is often converted into a log-likelihood function to replace expensive multiplication operations with additions. (This conversion is possible due to the fact that the logarithm is a monotonically increasing function of its argument and $p(\vec{x}_t\,|\,\lambda_i) \geq 0$.) The log-likelihood function is

$$\log\big(p(O\,|\,\lambda_i)\big) = \log\left(\prod_{t=1}^{T} p(\vec{x}_t\,|\,\lambda_i)\right) = \sum_{t=1}^{T} \log\big(p(\vec{x}_t\,|\,\lambda_i)\big) \ . \tag{11.15}$$

Assuming statistically independent feature vectors, the MAP decision rule becomes

$$\begin{aligned} \hat{\lambda}_{MAP} &= \arg\max_{\lambda_i}\left(P(\lambda_i)\cdot\prod_{t=1}^{T} p(\vec{x}_t\,|\,\lambda_i)\right) \\ &= \arg\max_{\lambda_i}\left(\log\big(P(\lambda_i)\big) + \sum_{t=1}^{T}\log\big(p(\vec{x}_t\,|\,\lambda_i)\big)\right) \quad i = 1, 2, \cdots c \ , \end{aligned} \tag{11.16}$$

and the ML decision rule becomes

$$\begin{aligned}\hat{\lambda}_{ML} &= \arg\max_{\lambda_i} \prod_{t=1}^{T} p(\vec{x}_t \mid \lambda_i) \\ &= \arg\max_{\lambda_i} \sum_{t=1}^{T} \log\left(p(\vec{x}_t \mid \lambda_i)\right) \quad i = 1, 2, \cdots c \ .\end{aligned} \tag{11.17}$$

To summarize, the ML decision rule can be interpreted as choosing the class λ_i with the highest likelihood score $p(O \mid \lambda_i)$. The MAP decision rule also utilizes the likelihood score. However, the MAP decision rule weights the contribution of each likelihood score $p(O \mid \lambda_i)$ with its corresponding prior probability $P(\lambda_i)$. In this manner, MAP assigns greater significance to likelihood scores for classes that are known to occur more often.

11.2 MIXTURE MODELING

A mixture model is a generative approach to sequence classification that seeks to model the underlying statistical distribution of a dataset. The basic goal of mixture modeling is to estimate parameters of class models λ_i that efficiently and accurately represent the likelihood function $p(\vec{x} \mid \lambda_i)$. One of the most basic forms of mixture modeling comes from modeling each class as a one-dimensional Gaussian. Given a single feature x, the mean μ and standard deviation σ are estimated for each class under the assumption of a Gaussian (normal) probability density function

$$p(x \mid \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right] \tag{11.18}$$

where the shorthand $x \sim N(\mu, \sigma)$ is commonly used. Although estimating the mean and standard deviation can be performed quite efficiently, one-dimensional Gaussian modeling is extremely limiting to many practical classification problems. Classification tasks often require a multitude of features to achieve robust performance in the presence of high noise levels, distortion, and other environmental factors. A one-dimensional distribution can only model a single classification feature, so a multi-dimensional distribution is needed for more precise modeling of complex feature sets.

Given a set of d random variables $\{x_1, x_2, \cdots, x_d\}$ (classification features), a d-dimensional random vector $\vec{x}$ (classification feature vector) is constructed such that

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix}. \tag{11.19}$$

Each individual feature serves as a dimension, and each feature vector is a single point in a d-dimensional vector space. The d-dimensional Gaussian probability density function is

$$p(\vec{x} \mid \vec{\mu}, \mathbf{\Sigma}) = \frac{1}{(2\pi)^{d/2} \left|\mathbf{\Sigma}\right|^{1/2}} \exp\left[-\frac{1}{2}(\vec{x} - \vec{\mu})^T \mathbf{\Sigma}^{-1} (\vec{x} - \vec{\mu})\right] \tag{11.20}$$

where the notation $(\cdot)^T$ stands for transpose, $|\cdot|$ is the matrix determinant, and $(\cdot)^{-1}$ is the matrix inverse. The traditional shorthand notation is $\vec{x} \sim N(\vec{\mu}, \mathbf{\Sigma})$ (that is, the random vector $\vec{x}$ has a multivariate normal distribution with mean vector $\vec{\mu}$ and covariance matrix $\mathbf{\Sigma}$). The mean vector $\vec{\mu}$ and the d-by-d covariance matrix $\mathbf{\Sigma}$ are

$$\vec{\mu} = E[\vec{x}] = E\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix} = \begin{bmatrix} E[x_1] \\ E[x_2] \\ \vdots \\ E[x_d] \end{bmatrix} \tag{11.21}$$

and

$$\mathbf{\Sigma} = E\left[(\vec{x}-\vec{\mu})(\vec{x}-\vec{\mu})^T \right] = \begin{bmatrix} \sigma_1^{\ 2} & \sigma_{1,2} & \cdots & \sigma_{1,d} \\ \sigma_{2,1} & \sigma_2^{\ 2} & & \vdots \\ \vdots & & \ddots & \\ \sigma_{d,1} & \cdots & & \sigma_d^{\ 2} \end{bmatrix} \tag{11.22}$$

where $E\left[\cdot\right]$ is the expectation operator and $\sigma_{i,j} = E\left[\left(x_i-\mu_i\right)\left(x_j-\mu_j\right)\right]$ is the covariance of feature x_i and feature x_j. Although the multivariate Gaussian allows for multi-dimensional feature sets, it relies on the assumption that the underlying distribution of the data is multivariate Gaussian. Classification performance may be severely degraded if the actual distribution of the data is not well modeled by a multivariate Gaussian. For example, if the distribution is multi-modal, a single multivariate Gaussian would be a poor choice.

The general form of a mixture model addresses the limitations of both the univariate and multivariate Gaussian models. A mixture density is a weighted sum of M component densities $p_i(\vec{x})$ such that

$$p(\vec{x}\,|\,\lambda) = \sum_{i=1}^{M} w_i p_i(\vec{x}) \tag{11.23}$$

where w_i are the mixture weights and λ represents the collective parameters of the mixture model. For $p(\vec{x}\,|\,\lambda)$ to be a valid density, each component density $p_i(\vec{x})$ must satisfy the following probability normalization condition

$$\int_{-\infty}^{\infty} p_i(\vec{x})d\vec{x} = 1 \qquad i = 1,\, 2,\, \cdots,\, M \tag{11.24}$$

and the mixture weights must sum to unity

$$\sum_{i=1}^{M} w_i = 1\,. \tag{11.25}$$

Provided these stochastic constraints are met, any arbitrary set of densities can be used in the mixture density.

Gaussian mixture models (GMMs) [82], [83], [84], [21], [85], [86], [87] utilize multivariate Gaussians as the component densities. A GMM satisfies the following equation

$$p\left(\vec{x} \mid \lambda\right) = \sum_{i=1}^{M} w_i N\left(\vec{\mu}_i, \mathbf{\Sigma}_i\right) \tag{11.26}$$

where shorthand notation has been used for clarity. Each GMM λ is represented by M mixture weights, M mean vectors, and M covariance matrices such that

$$\lambda = \left\{w_i, \vec{\mu}_i, \mathbf{\Sigma}_i\right\} \qquad i = 1,\ 2, \cdots,\ M\ . \tag{11.27}$$

Gaussian mixture models employ multivariate normal densities as basis functions to model arbitrary probability density functions. This is analogous to a Fourier series modeling arbitrary deterministic functions using sinusoidal basis functions. As the number of component densities increase, the better the model fits the given training data. However, in practical classification tasks, one must limit the number of model free parameters due to finite amounts of training data and computational resources.

The total number of free model parameters depends on both the dimensionality of the feature vector and the structure of the covariance matrix. The mixture weights contribute M free parameters, and the multivariate Gaussian mean vectors add $M \cdot d$ free parameters. The remaining free parameters are determined by the type of covariance matrix assumed. The three most common covariance matrix types used in GMMs are: full, spherical, and diagonal. The most general covariance matrix is the full $d \times d$ covariance matrix given by equation (11.22). A full covariance matrix allows covariance among all d features. The number of free parameters for a full covariance matrix is $M \cdot d^2$.

If the component Gaussians are restricted to have zero covariance among the features, the full covariance matrix reduces to the following diagonal covariance matrix

$$\mathbf{\Sigma}_i = \begin{bmatrix} \sigma_1^{\;2} & 0 & \cdots & 0 \\ 0 & \sigma_2^{\;2} & & \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & & \sigma_d^{\;2} \end{bmatrix}. \tag{11.28}$$

Diagonal covariance matrices restrict the major axes of the component Gaussians to be parallel to the coordinate axes. A diagonal covariance Gaussian mixture can still model an arbitrary distribution. However, a diagonal covariance GMM typically takes more component Gaussians to represent a distribution with the same accuracy as a full covariance GMM with a fewer number of component Gaussians. A diagonal covariance matrix adds $M \cdot d$ parameters to the total number of free GMM model parameters.

A diagonal covariance matrix reduces to a spherical covariance matrix under the additional assumption that all feature variances are equal such that

$$\mathbf{\Sigma}_i = \begin{bmatrix} \sigma^2 & 0 & \cdots & 0 \\ 0 & \sigma^2 & & \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & & \sigma^2 \end{bmatrix} = \sigma^2 \mathbf{I}_d \tag{11.29}$$

where $\mathbf{I}_d$ is the $d \times d$ identity matrix. With enough component Gaussians, a spherical covariance matrix GMM can also represent an arbitrary probability density function. The spherical covariance matrix adds only M free parameters. Table 11.1 summarizes the previous discussion.

Table 11.1. Total number of GMM free parameters

Covariance Matrix	M=64, d=7	M=64, d=12	M=128, d=7	M=128, d=14
Full (M+Md+Md^2)	3,648	10,048	7,296	20,096
Diagonal (M+Md+Md)	960	1,600	1,920	3,200
Spherical (M+Md+M)	576	896	1,152	1,792

Full, diagonal, or spherical covariance matrix GMMs can all model an arbitrary probability density function. However, practical considerations typically advocate the use of diagonal covariance matrix GMMs. While full covariance GMMs require the least number of component densities to achieve a given level of modeling accuracy, numerical problems often result from trying to estimate all $d \times d$ elements of the full covariance matrix. Limited training data often produces singularities in the inverse covariance matrix calculation. Diagonal covariance matrices do not suffer as much from singularity problems, and the number of component densities is also less than what is required for spherical covariance matrices.

11.3 GAUSSIAN MIXTURE MODEL CLASSIFICATION

Figure 11.1 illustrates the procedure for Gaussian mixture model classification.

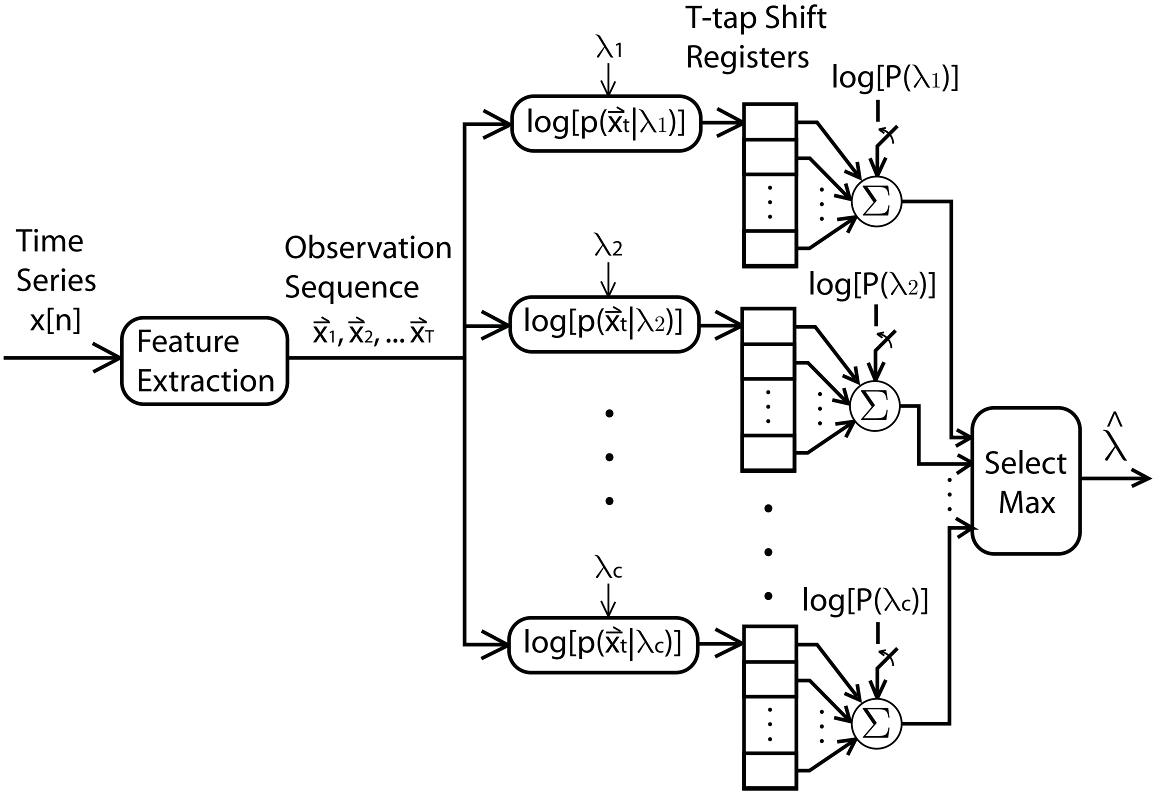


Figure 11.1. Gaussian mixture model sequence classification. The ML decision rule is implemented when the switches are open. Closing the switches implements the MAP decision rule.

Classification begins by extracting a sequence of feature vectors from the time series. The log-likelihood score is then computed for each of the c classes. The log-likelihood scores are based on the sum of T individual feature vector log-likelihoods (assuming independent sequential observations). As seen in Figure 11.1, the log-likelihood score is computed by summing the outputs of a T-tap shift register. The prior class probabilities are included when the switches in Figure 11.1 are closed, thus implementing the MAP decision rule. When the switches are open, the ML decision rule is implemented since the decision is made by choosing the class with the highest log-likelihood score.

Chapter 12: Classifier Training

12.1 TRAINING, PARAMETER TUNING, AND TESTING DATASETS

The micro-Doppler signature database is partitioned into three disjoint sets: the training set, parameter tuning set, and testing set. The training set is the dataset that is directly used by the classification algorithms to learn statistical models and/or decision regions. A large percentage (~66%) of the database is partitioned into the training set since classifier performance generally improves as the size of the training database increases.

A much smaller percentage (~12%) of the database is reserved for the parameter tuning set. The parameter tuning set is used to guide the feature selection process and optimize a classifier's free parameters. For example, a support vector machine's cost parameter C and radial basis function parameter γ are optimized by use of the parameter tuning set. All feature selection and parameter adjustment is performed on the parameter tuning dataset. The disjoint training, parameter tuning, and testing datasets provide the algorithm designer freedom to optimize parameters while not compromising the validity of the final test results. After all algorithm optimization is complete, the performance of the classifier is evaluated on the testing set. The testing set utilizes the remainder of the database (~22%).

This chapter focuses on classifier training. Therefore, all classification results shown in this chapter are those obtained on the *parameter tuning set* (except for Section 12.11 which uses subsets held-out from training set cross-validation). These results should be interpreted as *intermediate* test results to be used for classifier optimization only. The next chapter (Chapter 13) details classifier performance on the *testing set*. Chapter 13 reveals the actual predicted performance of the classifiers.

12.2 STRATIFIED DATABASE PARTITIONING

Partitioning a database into training, parameter tuning, and testing sets can be performed in several ways. Holdout [88] is one partitioning method that divides the database in a purely random fashion. No attention is paid to the distribution of the data examples among each partition. The problem with such a simple scheme is that each of the three partitions may not accurately represent the distribution of the database. If a particular class of data is under-represented in the database, it might only show up in one of the three partitions. If this data class appears only in the testing set, the classifier has had no chance to train for this data type and will likely perform poorly on that example. Similarly, if this data class only ends up in the training or parameter tuning sets, the classifier will not be tested against this data type (potentially resulting in optimistic performance estimates).

Stratified database partitioning (stratified holdout) attempts to remedy the problems of the holdout procedure. This approach first sub-divides the entire database into the major classes (e.g., human, vehicle, and animal). Then each class is further divided into smaller sub-classes (e.g., dog, bird, deer, goat, etc.). Each sub-category is then randomly split into training, parameter tuning, and testing sets according to the percent of samples allocated to each set. Stratified partitioning provides a more equal distribution of data samples among the three datasets. Table 12.1 illustrates the partitioning of the micro-Doppler database at the class level.

Table 12.1. Stratified database partitioning at the class level

	Training	Parameter Tuning	Testing	Sub-total	% of Total
Human	1315	229	434	1978	50.4
Vehicle	647	114	203	964	24.6
Animal	632	136	215	983	25.0
Sub-total	2954	479	852	3925	100.0
% of Total	66.1	12.2	21.7	100.0	

The human class comprises slightly over 50 percent of the entire database. The vehicle and animal classes each make up approximately 25 percent of the database. Table 12.1 illustrates how the distribution of the three classes is approximately equal in each of the training, parameter tuning, and testing datasets.

Tables 12.2, 12.3, and 12.4 show the division of the database into sub-classes. The main human sub-classes are: various approach angles, army crawling, and hands-and-knees baby crawling. The approach angle is the approximate angle that the target approaches or recedes from the radar antenna boresight. The “random” approach angle signifies datasets where the target changes its approach angle at random during the course of the data collection run. Each sub-class contains data at various ranges, speeds (walking, jogging, and running), and accelerations. The human class data came from five subjects ranging from 5 foot 4 inches tall to 6 foot 4 inches tall. Four male subjects and one female subject make up the human class.

Table 12.2. Stratified database partitioning of the human class

	Training	Parameter Tuning	Testing	Sub-total	% of Total
0°-30° approach ∠	862	140	255	1257	63.5
30°-60° approach ∠	102	16	34	152	7.7
90° approach ∠	72	14	26	112	5.7
Random approach ∠	67	17	41	125	6.3
Army crawl	147	28	51	226	11.4
Baby crawl	65	14	27	106	5.4
Sub-total	1315	229	434	1978	100.0
% of Total	66.5	11.6	21.9	100.0	

Data from a station wagon and a sports utility vehicle make up most of the vehicle data. In addition to these vehicles, data from random passing vehicles was also included in the database. Table 12.3 details the partitioning of the vehicle class. Although the primary break-down of the vehicle dataset is into approach angle, each vehicle sub-class included data at various ranges, speeds, and accelerations.

Table 12.3. Stratified database partitioning of the vehicle class

	Training	Parameter Tuning	Testing	Sub-total	% of Total
0°-30° approach ∠	420	70	129	619	64.2
30°-60° approach ∠	51	13	18	82	8.5
90° approach ∠	66	12	22	100	10.4
Random approach ∠	110	19	34	163	16.9
Sub-total	647	114	203	964	100.0
% of Total	67.1	11.8	21.1	100.0	

Table 12.4 shows the partitioning of the animal class. The animal dataset is subdivided into data from birds, deer, dogs, and goats. Doves, grackles, and mockingbirds were among the bird data. Data was collected from flying birds and birds walking on the ground. Large groups (over 15) of birds flying together are also part of the dataset. Deer are the most under-represented animal in the database. The deer were easily spooked by passing cars, humans, and other animals. In addition, much of the acquired deer datasets were taken near the maximum detectable range or the edge of the antenna beamwidth. Obtaining a deer dataset with good signal to noise ratio was rare. The dog data came from a 15-pound Yorkshire terrier. The dog chased a disk-shaped toy at different ranges and approach angles. In addition, the dog moved about randomly within the radar range. The goat datasets were acquired on a ranch in central Texas. These datasets included over 10 goats, many of which were simultaneously in motion during a data collection run.

Table 12.4. Stratified database partitioning of the animal class

	Training	Parameter Tuning	Testing	Sub-total	% of Total
Bird	279	48	87	414	42.1
Deer	55	9	19	83	8.4
Dog	202	34	64	300	30.5
Goat	96	45	45	186	18.9
Sub-total	632	136	215	983	100
% of Total	64.3	13.8	21.9	100	

12.3 SVM PERFORMANCE VERSUS THE COST AND GAMMA PARAMETERS

The radial basis function (RBF) was chosen for the support vector machine kernel. The RBF kernel is well-behaved mathematically, uses only a single free parameter γ, and performs well on many classification tasks. When using the RBF kernel, the SVM designer must provide appropriate values for the cost parameter C and the RBF kernel parameter γ. There is an infinite search space for these two parameters, but fortunately, small changes in these two parameters typically do not cause drastic performance variations. A common rule-of-thumb suggests that a good starting value for the kernel parameter γ is the inverse of the number of classification features. The results of a grid search over the cost parameter C and kernel parameter γ are shown in Table 12.5.

Table 12.5. Grid-search over SVM cost parameter C and kernel parameter γ. Performance is characterized by F-measure and accuracy (in parenthesis).

				γ			
		0.010	0.035	0.060	0.085	0.110	0.135
	256	84.1 (84.8)	89.8 (88.7)	90.6 (90.0)	91.6 (90.6)	91.9 (90.8)	92.6 (91.2)
	512	86.7 (86.2)	91.1 (90.0)	91.9 (90.8)	93.8 (92.3)	93.8 (92.3)	93.4 (91.9)
C	1024	90.3 (89.1)	92.8 (91.4)	94.2 (92.5)	93.4 (92.1)	93.0 (91.6)	93.4 (92.1)
	2048	92.6 (90.8)	94.2 (92.7)	93.2 (91.6)	93.5 (92.1)	93.1 (91.4)	93.0 (90.8)
	4096	92.8 (91.2)	93.6 (92.3)	93.5 (91.9)	93.0 (91.2)	93.0 (91.0)	93.0 (91.0)

The performance of the classifiers is quantified by the F-measure (and the 3-class accuracy in parenthesis). Except for small values of C and γ, the SVM performance is largely consistent over the search space. From Table 12.5, the pair of parameters $C = 2048$ and $\gamma = 0.035$ performs best.

12.4 NUMBER OF SUPPORT VECTORS VERSUS THE COST PARAMETER

The number of support vectors characterizes the complexity of the support vector machine classifier. The tradeoff between minimizing empirical risk and maximizing margin is highly dependent on the number of support vectors. Figure 12.1 shows the variation in the number of support vectors for each class as C is varied and γ is held constant at 0.035.

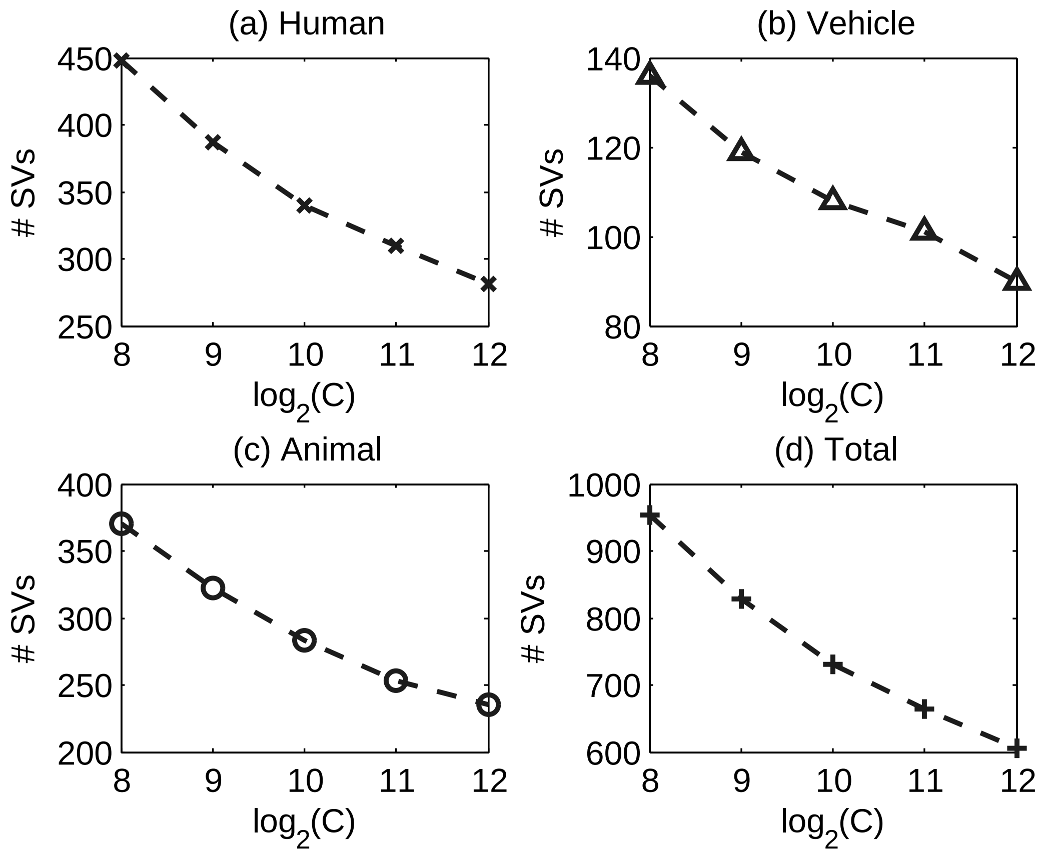


Figure 12.1. Number of support vectors versus C for human-versus-rest (a), vehicle-versus-rest (b), and animal-versus-rest (c) SVMs. Total number of support vectors in (d).

The number of support vectors decreases monotonically as C increases. As shown in Figure 12.1, the human-versus-rest and animal-versus-rest SVMs require significantly more support vectors than the vehicle-versus-rest SVM. This is due to the relative complexity of the human and animal classes versus the vehicle class (which contains primarily bulk-scatterer motion). The human class requires the most support vectors.

This observation is not unexpected as the human class includes examples of both biped (two-legged walking, jogging, running, etc.) and quadruped motion (army crawling and hands-and-knees crawling). Modeling a wide range of locomotion types requires more complex classifiers. However, overly complex classifiers typically perform poorly on novel datasets. Referring back to Table 12.5, the combination of $C = 2048$ and $\gamma = 0.035$ creates a classifier with a good balance of model complexity and generality.

12.5 GMM PERFORMANCE VERSUS THE NUMBER OF GAUSSIAN MIXTURES

The primary free parameters of the Gaussian mixture model (GMM) are the covariance matrix structure and the number of Gaussian mixtures M. The diagonal covariance matrix was chosen for numerical stability reasons as discussed in Chapter 11. There is no known analytical method for selecting the optimum value M for a given classification task. The GMM classifier designer typically trains the classifier using various values for M, and the parameter M is selected based on classifier performance on the parameter tuning set. The performance of the GMM classifier as a function of the number of Gaussian mixtures is shown in Figure 12.2.

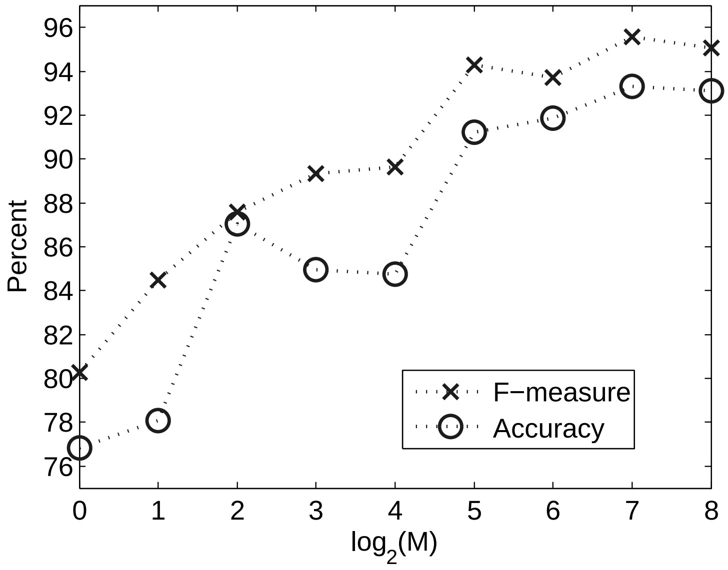


Figure 12.2. GMM performance versus the number of Gaussian mixtures M .

The GMM reached its peak performance when $M = 128$. The performance began to decrease at $M = 256$, and the computer ran out of memory at $M = 512$. Therefore, $M = 128$ was chosen for the GMM classifier.

12.6 PERFORMANCE OF SELECTED CLASSIFIERS

This section will analyze the performance of the chosen SVM and GMM classifiers in detail. The feature set utilized in this section is the Fisher score initialized sequential backward selection (FSISBS) feature set shown in Table 9.1. The selected SVM parameters are $C = 2048$ and $\gamma = 0.035$, and the GMM parameter is $M = 128$. The main performance metrics utilized are the confusion matrix, F-measure, and accuracy. The general performance of the classifier is discussed, followed by a performance analysis on target sub-classes.

Table 12.6 shows the performance of the SVM classifier on the parameter tuning set.

Table 12.6. SVM performance using FSISBS feature set

Predicted Class

		Human	Vehicle	Animal	% Error	% Correct
Actual Class	Human	212	8	9	7.4	92.6
	Vehicle	6	108	0	5.3	94.7
	Animal	11	1	124	8.8	91.2

F-measure	$P_{detection}$	$P_{nuisance-alarm}$	Accuracy
94.2%	97.4%	8.8%	92.7%

The SVM classifier shows good performance on the radar target classification task. The SVM simultaneously exhibits high detection probability and low nuisance alarm probability. Distinguishing human and animal targets is the most difficult task for the SVM classifier. Classifying biped humans from quadruped animals is a complex task of its own. Human targets undergoing quadruped motion (crawling) adds another level of complexity to the problem. In order to investigate the SVM classifier in more detail, Tables 12.7 through 12.9 show SVM performance on the sub-class level.

Table 12.7. SVM performance on human sub-classes

Human Sub-class	Human	Vehicle	Animal	% Error	% Correct
	Predicted Class				
Army Crawl	24	2	2	14.3	85.7
Baby Crawl	14	0	0	0.0	100.0
0°-30° ∠	135	5	0	3.6	96.4
30°-60° ∠	14	0	2	12.5	87.5
90° ∠	10	0	4	28.6	71.4
Random ∠	15	1	1	11.8	88.2

Table 12.8. SVM performance on vehicle sub-classes

Vehicle Sub-class	Human	Vehicle	Animal	% Error	% Correct
	Predicted Class				
0°-30° ∠	1	69	0	1.4	98.6
30°-60° ∠	1	12	0	7.7	92.3
90° ∠	3	9	0	25.0	75.0
Random ∠	1	18	0	5.3	94.7

Table 12.9. SVM performance on animal sub-classes

Animal Sub-class	Human	Vehicle	Animal	% Error	% Correct
	Predicted Class				
Bird	1	0	47	2.1	97.9
Deer	7	1	1	88.9	11.1
Dog	0	0	34	0.0	100.0
Goat	3	0	42	6.7	93.3

From Table 12.7, the SVM performs well on both the army crawl and baby crawl sub-classes. This result is encouraging since it reveals that the classifier is able to deal with significant changes in locomotion types. Both Table 12.7 and 12.8 illustrate feature set sensitivity to approach angle. The performance of the classifier gets progressively worse as the approach angle nears 90°. This result is not unexpected as the micro-Doppler response is produced by target motion *relative* to the radar. When the approach angle is approximately 90°, there is little relative motion between a target's limbs and the radar (and thus minimal micro-Doppler). The underlying feature set is highly dependent on the micro-Doppler response for its discriminatory power. Therefore, it is not surprising that the classifier performance varies with approach angle.

Table 12.9 shows that with the exception of the deer sub-class, the SVM performs well on each animal sub-class. The classifier does a poor job of discriminating deer from other classes. One explanation for this result is that the deer sub-class is highly under-represented in the training set and parameter tuning set. From Table 12.4, the deer class has only 55 examples in the training set and 9 examples in the parameter tuning set. The marginal amount of training data is a likely cause of the poor performance on deer targets. Collecting deer data was exceedingly difficult due to the nature of deer being easily spooked. The absence of a significant number of deer examples hindered the classifier's ability to discriminate deer from other targets.

The performance of the Gaussian mixture model classifier is displayed in Table 12.10.

Table 12.10. GMM performance using FSISBS feature set

		Predicted Class				
Actual Class		Human	Vehicle	Animal	% Error	% Correct
	Human	217	2	10	5.2	94.8
	Vehicle	12	102	0	10.5	89.5
	Animal	8	0	128	5.9	94.1

F-measure	$P_{detection}$	$P_{nuisance\text{-}alarm}$	Accuracy
95.6%	97.1%	5.9%	93.3%

The GMM classifier performs marginally better than the SVM classifier with regard to F-measure, accuracy, and nuisance-alarm probability. The GMM classifier has more difficulty in classifying the vehicle class than the SVM classifier. However, all of the GMM errors on the vehicle class are vehicles misclassified as humans (which does not influence the nuisance alarm probability). As with the SVM, the GMM classifier exhibits most of its error in distinguishing humans and animals.

The GMM performance at the sub-class level is shown in Table 12.11 through 12.13. Table 12.11 shows that the GMM classifier performs well on both army crawlers and baby crawlers. Table 12.11 and Table 12.12 show that the GMM classifier performance varies with approach angle, much like the SVM classifier. This observation further illustrates that the discriminatory power of the underlying feature set is dependent on approach angle. Table 12.13 reveals another similarity of the GMM and SVM classifiers. Both classifiers perform poorly on the under-represented deer sub-class but perform superbly on all the other animal sub-classes. The under-representation of the deer class in both the training and parameter tuning sets affects both classifiers' discrimination ability.

Table 12.11. GMM performance on human sub-classes

		Predicted Class				
		Human	Vehicle	Animal	% Error	% Correct
Human Sub-class	Army Crawl	26	0	2	7.1	92.9
	Baby Crawl	13	0	1	7.1	92.9
	0°-30° ∠	138	2	0	1.4	98.6
	30°-60° ∠	14	0	2	12.5	87.5
	90° ∠	11	0	3	21.4	78.6
	Random ∠	15	0	2	11.8	88.2

Table 12.12. GMM performance on vehicle sub-classes

		Predicted Class				
		Human	Vehicle	Animal	% Error	% Correct
Vehicle Sub-class	0°-30° ∠	1	69	0	1.4	98.6
	30°-60° ∠	7	6	0	53.8	46.2
	90° ∠	4	8	0	33.3	66.7
	Random ∠	0	19	0	0.0	100.0

Table 12.13. GMM performance on animal sub-classes

		Predicted Class				
		Human	Vehicle	Animal	% Error	% Correct
Animal Sub-class	Bird	0	0	48	0.0	100.0
	Deer	7	0	2	77.8	22.2
	Dog	0	0	34	0.0	100.0
	Goat	1	0	44	2.2	97.8

12.7 CLASSIFIER PERFORMANCE USING ONLY STATISTICAL FEATURES

Section 12.6 detailed classifier performance using the feature set in Table 9.1. Recall that the feature set of Table 9.1 was automatically selected using a Fisher score initialized sequential backward selection (FSISBS) algorithm. Sections 12.7 through 12.9 will analyze classifier effectiveness using various subsets of the feature set in Table 9.1.

This section analyzes SVM and GMM performance using only "statistical features." The statistical features include all of the features from Table 9.1 except for micro-Doppler period, range-weighted target energy, and micro-Doppler energy. The set of statistical features have a greater range of applicability than the heuristic features. For example, the range-weighted target energy requires a sensor that provides target range estimates. This requirement precludes the use of this feature in a classifier based on a simple continuous wave (CW) radar. Table 12.14 and Table 12.15 illustrate SVM and GMM performance utilizing only statistical features. The elimination of heuristic features has considerably increased the nuisance alarm probability and decreased the F-measure for both SVM and GMM. Tables 12.14 and 12.15 highlight the importance of the heuristic features in obtaining high-performance classification.

Table 12.14. SVM performance using only statistical features

Predicted Class

Actual Class	Human	Vehicle	Animal	% Error	% Correct
Human	187	16	26	18.3	81.7
Vehicle	9	102	3	10.5	89.5
Animal	58	8	70	48.5	51.5

F-measure	$P_{detection}$	$P_{nuisance\text{-}alarm}$	Accuracy
65.9%	91.6%	48.5%	75.0%

Table 12.15. GMM performance using only statistical features

Predicted Class

Actual Class	Human	Vehicle	Animal	% Error	% Correct
Human	140	26	63	38.9	61.1
Vehicle	21	73	20	36.0	64.0
Animal	50	2	84	38.2	61.8

F-measure	$P_{detection}$	$P_{nuisance\text{-}alarm}$	Accuracy
68.1%	75.8%	38.2%	62.0%

12.8 CLASSIFIER PERFORMANCE USING ONLY HEURISTIC FEATURES

This section assesses the effectiveness of the heuristic feature set. Thus, the micro-Doppler period, range-weighted target energy, and micro-Doppler energy are the only features used to train the classifier. Table 12.16 and Table 12.17 show the SVM and GMM classification results using only heuristic features. Both classifiers are seen to perform better on the strictly heuristic feature set than the strictly statistical feature set. This result emphasizes the discrimination ability of the heuristic feature set considering that the heuristic-only set contained three features while the statistical-only feature set contained nine features. However, the combined heuristic and statistical feature set outperformed the heuristic-only feature set in every metric.

Table 12.16. SVM performance using only heuristic features

Predicted Class

		Human	Vehicle	Animal	% Error	% Correct
Actual Class	Human	195	4	30	14.8	85.2
	Vehicle	13	101	0	11.4	88.6
	Animal	30	0	106	22.1	77.9

F-measure	$P_{detection}$	$P_{nuisance\text{-}alarm}$	Accuracy
84.1%	91.2%	22.1%	83.9%

Table 12.17. GMM performance using only heuristic features

Predicted Class

		Human	Vehicle	Animal	% Error	% Correct
Actual Class	Human	191	10	28	16.6	83.4
	Vehicle	8	104	2	8.8	91.2
	Animal	17	0	119	12.5	87.5

F-measure	$P_{detection}$	$P_{nuisance\text{-}alarm}$	Accuracy
89.3%	91.2%	12.5%	86.4%

12.9 CLASSIFIER PERFORMANCE WITHOUT THE USE OF RANGE INFORMATION

This section analyzes classifier performance without the use of range information. This analysis is performed to give an estimate of the feature set and classifier performance when applied to a simple continuous wave (CW) Doppler radar (instead of the MFCW radar used in this dissertation). The only feature based on a range estimate is the range-weighted target energy. The feature set used in this section contains all of the features of Table 9.1, except for range-weighted target energy.

Table 12.18 and Table 12.19 display SVM and GMM performance without range-based features. Comparing Tables 12.18 and 12.19 to Tables 12.6 and 12.10 (respectively) reveals the importance of the range estimate. Every classification metric is degraded for both SVM and GMM when the single range-based feature is removed from the feature set. While the detection probability is not highly degraded, the probability of nuisance alarm noticeably increases. The GMM classifier out-performs the SVM classifier when the feature set contains no range information.

Table 12.18. SVM performance without the use of range information

		Predicted Class				
		Human	Vehicle	Animal	% Error	% Correct
Actual Class	Human	204	11	14	10.9	89.1
	Vehicle	10	103	1	9.6	90.4
	Animal	41	7	88	35.3	64.7

F-measure	$P_{detection}$	$P_{nuisance\text{-}alarm}$	Accuracy
77.2%	95.6%	35.3%	82.5%

Table 12.19. GMM performance without the use of range information

		Predicted Class				
		Human	Vehicle	Animal	% Error	% Correct
Actual Class	Human	204	10	15	10.9	89.1
	Vehicle	10	98	6	14.0	86.0
	Animal	28	2	106	22.1	77.9

F-measure	$P_{detection}$	$P_{nuisance-alarm}$	Accuracy
85.2%	93.9%	22.1%	85.2%

12.10 CLASSIFIER PERFORMANCE USING THE FISHER SCORE FEATURE SET

The FSISBS feature selection algorithm (as described in Chapter 9) used the Fisher score as an initialization step followed by sequential backward selection (SBS). The Fisher score initialization retains features with the highest (one-dimensional) discriminative power and reduces the computational requirements of the following SBS step. The SBS step is used to remove features that contain redundant information from the feature set.

The Fisher score initialization retained 33 of the original 80 extracted features. SBS further reduced the number of features from 33 to 12. The classifiers in this section are trained with the 33 features selected by Fisher score initialization. This test is performed to asses the effectiveness of the SBS step and to analyze each classifier's discrimination ability in the presence of correlated features. Table 12.20 and Table 12.21 show the performance of the SVM and GMM classifiers using the 33 features selected solely by the one-dimensional Fisher score.

Table 12.20. SVM performance using feature set selected by Fisher score only

Actual Class	Predicted Class Human	Vehicle	Animal	% Error	% Correct
Human	214	8	7	6.6	93.4
Vehicle	4	110	0	3.5	96.5
Animal	14	1	121	11.0	89.0

F-measure	$P_{detection}$	$P_{nuisance\text{-}alarm}$	Accuracy
93.2%	98.0%	11.0%	92.9%

Table 12.21. GMM performance using feature set selected by Fisher score only

Actual Class	Predicted Class Human	Vehicle	Animal	% Error	% Correct
Human	154	3	72	32.8	67.2
Vehicle	5	102	7	10.5	89.5
Animal	1	0	135	0.7	99.3

F-measure	$P_{detection}$	$P_{nuisance\text{-}alarm}$	Accuracy
86.7%	77.0%	0.7%	81.6%

A comparison of Table 12.20 with Table 12.6 indicates that SVM classifier performance is only mildly degraded by eliminating the sequential backwards selection (SBS) step. The nuisance alarm probability is increased from 8.8% to 11.0% while the F-measure decreased from 94.2% to 93.2%. The results for GMM are quite different as seen by comparing Table 12.21 with Table 12.10. Although the nuisance alarm probability decreased from 5.9% to 0.7%, the F-measure, probability of detection, and accuracy were all considerably degraded. Many of the additional errors are attributed to human targets misclassified as animals. Without the SBS step, 37 of the 42 crawling human targets were misclassified as animals. The percent correct classification rate on crawling targets dropped from 92.9% to 11.9% by eliminating the SBS step.

Several interesting conclusions can be drawn from this test. First, the sequential backward selection algorithm improved classifier performance through feature set pruning. The GMM classifier performs considerably worse with 33 features than with 12 features. This result is a practical example of the famous “curse of dimensionality.” Adding extra features does not always result in better performance when a finite size training set is used. This test also revealed that the SVM classifier is more tolerant of redundant information within the feature set than the GMM classifier. The SVM classifier is better able to cope with the high dimensionality and redundancy of the feature set.

12.11 STATISTICAL SIGNIFICANCE TESTING USING CROSS-VALIDATION

Section 12.6 reported classification results for classifiers trained using the feature set selected by Fisher score initialized sequential backward selection (FSISBS). Section 12.10 compared these results to classifiers trained on a feature set selected solely by Fisher score initialization. SVM and GMM performance was shown to be quite similar for the FSISBS feature set. However, when only the Fisher score was used for feature selection, SVM and GMM performance differed markedly. The goal of this section is to determine if the performance differences between SVM and GMM are *statistically significant*.

Statistical significance calculations require multiple evaluation sets for sample statistic calculations. The pattern recognition community commonly uses n-fold cross-validation [88], [89] to provide the required evaluation sets. In n-fold cross-validation, the training set is randomly split into n disjoint folds (subsets) of equal size. The classifiers are trained using $n-1$ folds, and the classification performance is evaluated on the remaining fold. The process is repeated until the classifier performance has been evaluated on all folds.

Table 12.21 shows the results of 10-fold cross-validation using the FSISBS feature set. Classification accuracy, sample statistics, confidence intervals, and significance testing results are displayed in Table 12.21. Section 9.6 through Section 9.10 detail the theory behind the calculations displayed in Table 12.21. For the FSISBS feature set, SVM and GMM performance is remarkably similar. The average accuracy difference is less than 1%. The high amount of overlap between the SVM and GMM confidence intervals suggests that the performance difference is not statistically significant. This conclusion is further supported by the results significance testing. The paired Student's t-test resulted in $p = 0.161$. This $p-\text{value}$ is far too high to reject the null hypothesis. Thus, the paired Student's t-test also suggests that there is no statistically significant difference between the accuracy of the SVM and GMM classifiers when the FSISBS feature set is used.

Table 12.21. Sample statistics, confidence intervals, and paired Student's t-test using FSISBS feature set

Fold #	SVM Accuracy (%)	GMM Accuracy (%)	Difference, d (%)
1	94.6	94.2	0.4
2	93.8	91.9	1.9
3	92.3	91.9	0.4
4	91.5	91.5	0.0
5	90.0	89.6	0.4
6	90.7	93.0	-2.3
7	93.8	93.4	0.4
8	92.7	91.9	0.8
9	93.0	91.1	1.9
10	95.4	92.8	2.6
Sample Mean $\hat{\mu}_X$	92.8	92.1	$\hat{\mu}_D = 0.65$
Sample Standard Deviation $\hat{\sigma}_X$	1.71	1.31	$\hat{\sigma}_D = 1.35$
95% Confidence Interval $(\alpha = 0.05)$	$91.6 < \mu_X < 94.0$	$91.2 < \mu_X < 93.0$	$-0.31 < \mu_D < 1.61$ (50.0% overlap)
99% Confidence Interval $(\alpha = 0.01)$	$91.0 < \mu_X < 94.5$	$90.8 < \mu_X < 93.5$	$-0.73 < \mu_D < 2.03$ (67.6% overlap)
Degrees of Freedom r			9
Student's t-value t_r			1.53
Student's p-value p			0.161

Table 12.22 shows the results of 10-fold cross-validation using the feature set selected solely by the Fisher score. SVM and GMM performance differs considerably when the Fisher score is the only means of feature selection. Even at the 99% confidence level, no overlap occurs between the SVM and GMM confidence intervals. This result provides strong evidence that the difference in SVM and GMM accuracy is statistically significant. Significance testing agrees with this conclusion. The paired Student’s t-test resulted in $p = 3.78 \times 10^{-5}$. This quite low $p-$value supports the rejection of the null hypothesis. Hence, the paired Student’s t-test suggests that there is a statistically significant difference between the accuracy of the SVM and GMM classifiers when using the feature set selected only by the Fisher score.

The results of this section emphasized the importance of feature selection for high-performance classification. The next chapter analyzes the results of applying the selected SVM and GMM classifiers (as configured in Section 12.6 with the FSISBS feature set) to the testing set.

Table 12.22. Sample statistics, confidence intervals, and paired Student's t-test using Fisher score feature set

Fold #	SVM Accuracy (%)	GMM Accuracy (%)	Difference, d (%)
1	91.5	85.7	5.8
2	95.8	88.4	7.4
3	91.5	84.9	6.6
4	93.0	85.3	7.7
5	89.2	86.5	2.7
6	93.0	84.2	8.8
7	94.2	90.4	3.8
8	92.7	84.6	8.1
9	95.0	86.1	8.9
10	96.2	82.1	14.1
Sample Mean $\hat{\mu}_X$	93.2	85.8	$\hat{\mu}_D = 7.39$
Sample Standard Deviation $\hat{\sigma}_X$	2.16	2.29	$\hat{\sigma}_D = 3.12$
95% Confidence Interval $(\alpha = 0.05)$	$91.7 < \mu_X < 94.8$	$84.2 < \mu_X < 87.4$	$5.15 < \mu_D < 9.63$ (0.0% overlap)
99% Confidence Interval $(\alpha = 0.01)$	$91.0 < \mu_X < 95.4$	$83.5 < \mu_X < 88.2$	$4.18 < \mu_D < 10.6$ (0.0% overlap)
Degrees of Freedom r			9
Student's t-value t_r			7.48
Student's p-value p			3.78×10^{-5}

Chapter 13: Classifier Performance Evaluation

13.1 INTRODUCTION

The previous chapter detailed the process of classifier training and parameter tuning. The classifiers were trained using the training sets, and the parameters were tuned by assessing classifier performance on the parameter tuning set. Classification results on the parameter tuning set are typically optimistic since classifier parameters were altered based on results from the parameter tuning set. For this reason, the testing set (which is disjoint from both the training set and parameter tuning set) provides an independent test of classifier performance. The classifier is tested on the testing set only after all classifier parameter tuning has ceased.

This chapter analyzes the performance of the SVM and GMM classifiers on the testing set. Both the SVM and GMM classifiers use the Fisher score initialized sequential backward selection (FSISBS) feature set shown in Table 9.1. The SVM classifier uses the parameters $C = 2048$ and $\gamma = 0.035$. The GMM classifier uses the parameter $M = 128$.

13.2 CLASSIFICATION RESULTS

Table 13.1 shows the performance of the SVM classifier on the testing set. As expected, the results on the parameter tuning set were slightly optimistic (see Table 12.6). Each classification metric was degraded in the range of 2.6% to 3.2% when comparing the parameter tuning set results to the testing set results. However, SVM performance on the testing set is still quite good. Similar results on the parameter tuning and testing sets suggest that the training process learned appropriate decision boundaries and did not over-train the classifier.

Table 13.1. SVM performance on testing set

		Predicted Class				
Actual Class		Human	Vehicle	Animal	% Error	% Correct
	Human	397	5	32	8.5	91.5
	Vehicle	17	181	5	10.8	89.2
	Animal	17	8	190	11.6	88.4

F-measure	$P_{detection}$	$P_{nuisance\text{-}alarm}$	Accuracy
91.2%	94.2%	11.6%	90.1%

Tables 13.2 through 13.4 display SVM classification results on the testing set at the sub-class level. Table 13.2 shows SVM classification results on the human sub-class. The SVM classifier results on the testing set continue to show excellent performance on both types of crawling targets. However, the classifier performs better on baby crawlers than army crawlers. Table 13.2 further illustrates the variation of classifier performance as a function of approach angle. The lowest performing human sub-class is humans moving at a 90° approach angle. The minimal micro-Doppler produced at a 90° approach angle (coupled with a reduction in received signal strength from bulk scatterers) creates significant intra-class feature variability. This variability causes many of the 90° approach angle human targets to be misclassified as animals.

Table 13.3 shows that the vehicle sub-class is similarly affected by varying approach angle. Like the human class, the performance of the vehicle class is most degraded for targets at a 90° approach angle. As the vehicle approach angle nears 90°, the bulk-scatterer Doppler response spreads in frequency and the received signal strength decreases. These effects cause intra-class feature variability which results in increased vehicle misclassifications.

Table 13.2. SVM performance on human sub-classes (testing set)

	Predicted Class				
Human Sub-class	Human	Vehicle	Animal	% Error	% Correct
Army Crawl	46	0	5	9.8	90.2
Baby Crawl	27	0	0	0.0	100.0
0°-30° ∠	249	5	1	2.4	97.6
30°-60° ∠	28	0	6	17.6	82.4
90° ∠	11	0	15	57.7	42.3
Random ∠	26	0	5	12.2	87.8

Table 13.3. SVM performance on vehicle sub-classes (testing set)

	Predicted Class				
Vehicle Sub-class	Human	Vehicle	Animal	% Error	% Correct
0°-30° ∠	4	125	0	3.1	96.9
30°-60° ∠	3	15	0	16.7	83.3
90° ∠	8	11	3	50.0	50.0
Random ∠	2	30	2	11.8	88.2

Table 13.4. SVM performance on animal sub-classes (testing set)

	Predicted Class				
Animal Sub-class	Human	Vehicle	Animal	% Error	% Correct
Bird	4	0	83	4.6	95.4
Deer	8	8	3	84.2	15.8
Dog	3	0	61	4.7	95.3
Goat	2	0	43	4.4	95.6

Table 13.4 displays SVM performance on the testing set for the animal sub-classes. The results on the testing set are similar to the results on the parameter tuning

set. With the exception of the deer sub-class, SVM performance on the animal sub-class is quite good. The classifier performance on the bird, dog, and goat sub-classes is remarkably similar. As mentioned previously, poor performance on the deer sub-class is attributed to the under-representation of the deer sub-class in both the training and parameter tuning sets. The deer sub-class contained approximately half of the training examples and approximately one quarter of the parameter tuning examples of any of the other animal sub-classes. Half of the errors on the deer sub-class were due to deer being classified as vehicles. The amount of training examples provided for classifier training and parameter tuning was not enough to appropriately model the deer sub-class.

Table 13.5 displays GMM classifier performance on the testing set. As with the SVM classifier, the results for the GMM classifier on the parameter tuning set were slightly optimistic. A reduction in performance occurred across all classification metrics in the range of around 2.8% to 3.5%. However, the results on the testing set are still impressive.

Table 13.5. GMM performance on testing set

Predicted Class

Actual Class	Human	Vehicle	Animal	% Error	% Correct
Human	394	6	34	9.2	90.8
Vehicle	14	182	7	10.3	89.7
Animal	15	5	195	9.3	90.7

F-measure	$P_{detection}$	$P_{nuisance-alarm}$	Accuracy
92.1%	93.6%	9.3%	90.5%

Tables 13.6 through 13.8 show GMM performance on the testing set at the sub-class level. Table 13.6 and Table 13.7 further illustrate that classification performance varies with approach angle. The underlying feature set relies heavily on the micro-Doppler phenomenon which is dependent on approach angle. Table 13.8 shows the good performance of the GMM classifier on the animal sub-class, with the exception of the deer sub-class.

Table 13.6. GMM performance on human sub-classes (testing set)

Predicted Class

Human Sub-class	Human	Vehicle	Animal	% Error	% Correct
Army Crawl	49	0	2	3.9	96.1
Baby Crawl	27	0	0	0.0	100.0
0°-30° ∠	245	6	4	3.9	96.1
30°-60° ∠	27	0	7	20.6	79.4
90° ∠	8	0	18	69.2	30.8
Random ∠	38	0	3	7.3	92.7

Table 13.7. GMM performance on vehicle sub-classes (testing set)

Predicted Class

Vehicle Sub-class	Human	Vehicle	Animal	% Error	% Correct
0°-30° ∠	2	127	0	1.6	98.4
30°-60° ∠	3	13	2	27.8	72.2
90° ∠	6	11	5	50.0	50.0
Random ∠	3	31	0	8.8	91.2

Table 13.8. GMM performance on animal sub-classes (testing set)

Predicted Class

Animal Sub-class	Human	Vehicle	Animal	% Error	% Correct
Bird	0	0	87	0.0	100.0
Deer	11	5	3	84.2	15.8
Dog	1	0	63	1.6	98.4
Goat	3	0	42	6.7	93.3

13.3 CONFIDENCE INTERVALS AND STATISTICAL SIGNIFICANCE TESTING

The previous section provided detailed SVM and GMM performance results using the entire testing set. In order to calculate sample statistics and confidence intervals for classification accuracy, the classifier must be run multiple times on disjoint subsets of the testing set. In this section, the testing set is partitioned into 10 randomly selected (without replacement) subsets of equal size. Note that this is not a cross-validation procedure (as done in Section 12.11). The classifier applied to each subset of the testing set is exactly the same as the classifier of Section 13.2 (which is the same classifier of Section 12.6). Application of the classifier to smaller subsets is done to provide sample statistics and confidence intervals for classification accuracy.

Table 13.9 shows the results of applying the SVM and GMM classifiers to subsets of the testing set. Table 13.9 displays classification accuracy, sample statistics, confidence intervals, and significance testing results. SVM and GMM performance on the testing set is shown to be quite similar. The average difference in accuracy between the SVM and GMM classifiers is less than 1%. The SVM and GMM confidence intervals overlap considerably, which strongly suggests that the performance difference is not statistically significant. Significance testing further corroborates this conclusion. The paired Student's t-test produced $p = 0.604$. This $p-$value is quite high, which implies that the null hypothesis should not be rejected. These results suggest that the SVM and GMM accuracy difference on the testing set is not statistically significant.

Table 13.9. Testing set sample statistics, confidence intervals, and paired Student's t-test

Subset #	SVM Accuracy (%)	GMM Accuracy (%)	Difference, d (%)
1	90.6	90.6	0.0
2	92.9	91.8	1.1
3	91.8	95.3	-3.5
4	87.1	89.4	-2.3
5	90.6	94.1	-3.5
6	90.6	90.6	0.0
7	90.6	88.2	2.4
8	88.2	87.1	1.1
9	95.3	94.1	1.2
10	83.9	83.9	0.0
Sample Mean $\hat{\mu}_X$	90.1	90.5	$\hat{\mu}_D = -0.35$
Sample Standard Deviation $\hat{\sigma}_X$	3.16	3.53	$\hat{\sigma}_D = 2.06$
95% Confidence Interval $(\alpha = 0.05)$	$87.9 < \mu_X < 92.4$	$88.0 < \mu_X < 93.0$	$-1.82 < \mu_D < 1.12$ (86.3% overlap)
99% Confidence Interval $(\alpha = 0.01)$	$86.9 < \mu_X < 93.4$	$86.9 < \mu_X < 94.1$	$-2.47 < \mu_D < 1.77$ (90.3% overlap)
Degrees of Freedom r			9
Student's t-value t_r			-0.537
Student's p-value p			0.604

13.4 ANALYSIS OF CLASSIFICATION RESULTS

Many conclusions can be drawn from the results of Chapter 12 and Chapter 13. One such conclusion is that the classification results are more sensitive to feature selection than the choice of classification algorithm. The performance difference between the SVM and GMM classifiers using the chosen FSISBS feature set is less than 1%. Performance differences between a poorly chosen feature set and an appropriately selected feature set can be over an order of magnitude larger.

Sections 12.7 and 12.8 revealed that the classifiers performed best when a combination of statistical and heuristic features were used. However, the feature set consisting of three heuristic features performed better than a feature set containing nine statistical features. Section 12.9 highlighted the importance of range information to the classification results. The large majority of the selected features can be applied to target classification using a simple continuous wave radar (instead of an MFCW radar), albeit at a reduced performance level. The results of this dissertation also reveal that the GMM classifier is more sensitive to redundant features than the SVM classifier. The SVM classifier is seen to perform well even when using a feature set containing many redundant features.

This dissertation revealed that both the SVM and GMM classification performance varied as a function of approach angle. The sensitivity to approach angle is a direct result of the dependence of the feature set on the micro-Doppler phenomenon (which depends on approach angle). The sub-class most affected by the sensitivity to approach angle is the 90° approach angle human sub-class. Multiple “looks” may be needed to reliably detect a human target at or near 90° approach angle.

The overall performance of both the SVM and GMM classifiers is impressive. The classifiers perform well under variation in target speed, range, approach angle, and posture. Both the SVM and GMM classifiers deal well with deliberate attempts to spoof the classifier. The classification results on both army crawlers and hands-and-knees crawlers are quite good. Another encouraging result is that the classification performance is similar for the parameter tuning set and the testing set. The results on the parameter tuning set were only mildly optimistic—the maximum classification metric differential was 3.5%. When using the FSISBS feature set, the difference in SVM and GMM classification accuracy is not statistically significant.

Chapter 14: Conclusion

14.1 DISSERTATION CONTRIBUTIONS

This dissertation research achieved its primary scientific objective to perform robust, automatic detection and classification of moving non-cooperative targets using a sensor on a stationary platform. In addition, this dissertation research completed the other scientific objectives to design low-cost sensor hardware, collect and analyze an extensive radar signature database, and research the micro-Doppler phenomenon. This section reviews the contributions made by this dissertation to the body of knowledge.

A major contribution of this dissertation was the collection, processing, and analysis of a diverse micro-Doppler signature database. The database was collected over a three-year period (2004-2007) at multiple testing locations. The micro-Doppler signature database included datasets with targets moving at numerous ranges, velocities, and approach angles. The target classes included were human, vehicle, and animal. Bird, goat, deer, and crawling human micro-Doppler signatures were among the many novel datasets collected for this dissertation research. Evaluation of theory on realistic experimental data is vital to the advancement of knowledge. The data can be used to rigorously evaluate new classification, detection, and feature selection algorithms. Computer simulation analysis also plays a crucial role in theoretical development. The experimental data collected by this dissertation research can be utilized to improve the accuracy of computer models.

The design and testing of a low-cost multiple frequency continuous wave (MFCW) radar was contributed by this dissertation. The MFCW radar provides high performance target detection and classification at reasonable cost and complexity. Due to the balance of cost, complexity, and performance, the MFCW radar is a practical sensor for both industrial and academic applications. This dissertation also contributed a novel method for visualizing and extracting range estimates from MFCW radars: the time-frequency-range diagram (TFRgram).

This dissertation contributed a novel set of high-performance micro-Doppler based features. The Fisher score initialized sequential backward selection (FSISBS) algorithm selected a feature set with a combination of the micro-Doppler period, range-weighted target energy, and micro-Doppler energy heuristic features. In addition, the FSISBS feature set included both static and dynamic cepstral coefficients, cepstral energy features, and the linear predictive coding (LPC) residual energy feature. The FSISBS feature set was shown to perform well on the micro-Doppler based target classification problem.

The design and detailed analysis of target classification algorithms based on micro-Doppler features was contributed by this dissertation. Both support vector machine (SVM) and Gaussian mixture model (GMM) classifiers were designed to accomplish high-performance micro-Doppler based target classification. The importance of both classifier selection and feature selection was analyzed in detail. When using the feature set selected by the FSISBS algorithm, the performance difference between the SVM and GMM classifiers was not statistically significant. However, a statistically significant difference between the two classifiers was observed when using the feature set selected solely by the one-dimensional Fisher score. Both SVM and GMM classifiers are well suited to the radar target classification task.

14.2 FUTURE WORK

Although the scientific objectives of this dissertation were met, more research can be performed to further improve our knowledge of micro-Doppler based classification. One of the first priorities for continued research would be collecting more micro-Doppler datasets. Acquiring more data is important for several reasons. First, collecting a wider variety of target types improves our understanding of micro-Doppler. Target micro-Doppler responses can be used to improve radar simulation models and to study various electromagnetic scattering phenomena. In addition to increasing our knowledge of micro-Doppler, expanding the micro-Doppler signature database is vital for improving classification results. As shown by the classification results on deer, adequate target representation in the training set and parameter tuning set is crucial for high performance classification. Also, increasing the size and diversity of the testing set makes estimates of classifier performance more reliable.

The results of this dissertation support the argument that feature selection is often more critical than classifier selection. Classification performance is fundamentally limited by the inherent separability of the feature set. The Fisher score initialized sequential backward selection algorithm utilized in this dissertation performed well. However, other feature selection algorithms such as those incorporating genetic algorithms (GA) may perform better. In addition to feature selection, improving feature extraction is also vital for obtaining better results. Perhaps the most important feature to add would be an estimate of target approach angle. The target approach angle feature could be used to partially compensate for the angular dependence of micro-Doppler. Unfortunately, implementing the target approach angle feature would require significant changes to the current radar hardware since a minimum of one extra antenna and receive channel would be required.

References

[1] K. Murphy, "Hidden Markov model (HMM) toolbox for Matlab," Software available at: http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html, 2005.

[2] C. C. Chang and C. J. Lin, "LIBSVM: a library for support vector machines," Software available at: http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2001.

[3] M. A. Solano, J. S. Ipina, J. M. Zamanillo, and C. Perez-Vega, "X-band Gunn diode oscillator for a multiple-frequency continuous-wave radar for educational purposes," *Education, IEEE Transactions on,* vol. 45, pp. 316-322, November 2002.

[4] M. G. Anderson and R. L. Rogers, "Micro-Doppler analysis of multiple frequency continuous wave radar signatures," in *Radar Sensor Technology XI, Proceedings of SPIE*, Orlando, FL, USA, 2007, pp. 65470A-10.

[5] V. C. Chen and R. D. Lipps, "Time frequency signatures of micro-Doppler phenomenon for feature extraction," in *Wavelet Applications VII, Proceedings of SPIE*, Orlando, FL, USA, 2000, pp. 220-226.

[6] J. L. Geisheimer, E. F. Greneker III, and W. S. Marshall, "High-resolution Doppler model of the human gait," in *Radar Sensor Technology and Data Visualization, Proceedings of SPIE*, Orlando, FL, USA, 2002, pp. 8-18.

[7] P. van Dorp and F. C. A. Groen, "Human walking estimation with radar," *Radar, Sonar and Navigation, IEE Proceedings,* vol. 150, pp. 356-365, October 2003.

[8] J. Li and H. Ling, "Application of adaptive chirplet representation for ISAR feature extraction from targets with rotating parts," *Radar, Sonar and Navigation, IEE Proceedings,* vol. 150, pp. 284-91, August 2003.

[9] J. L. Geisheimer, W. S. Marshall, and E. F. Greneker III, "A continuous-wave (CW) radar for gait analysis," in *Signals, Systems and Computers, 2001. Conference Record of the Thirty-Fifth Asilomar Conference on*. vol. 1, 2001, pp. 834-838.

[10] P. Setlur, M. Amin, and F. Ahmad, "Analysis of micro-Doppler signals using linear FM basis decomposition," in *Radar Sensor Technology X, Proceedings of SPIE*, Orlando, FL, USA, 2006, pp. 62100M-11.

[11] V. C. Chen, "Spatial and temporal independent component analysis of micro-Doppler features," in *Radar Conference, 2005 IEEE International*, 2005, pp. 348-353.

[12] T. Thayaparan, S. Abrol, E. Riseborough, L. Stankovic, D. Lamothe, and G. Duff, "Analysis of radar micro-Doppler signatures from experimental helicopter and human data," *Radar, Sonar & Navigation, IET,* vol. 1, pp. 289-299, August 2007.

[13] W. S. Marshall, J. L. Geisheimer, E. Greneker, D. Fry, R. Imamura, and B. Johnson, "Radar-based gait measurement for human identification," in *Auto ID, IEEE Conference On* Tarrytown, NY, 2002.

[14] P. G. Kealey and M. Jahangir, "Advances in Doppler recognition for ground moving target indication," in *Automatic Target Recognition XVI, Proceedings of SPIE*, Orlando, FL, USA, 2006, pp. 62340W-11.

[15] A. Lin and H. Ling, "Through-wall measurements of a Doppler and direction-of-arrival (DDOA) radar for tracking indoor movers," in *Antennas and Propagation Society International Symposium, 2005 IEEE*, 2005, pp. 322-325 vol. 3B.

[16] R. Tan and R. Bender, "Analysis of Doppler measurements of people," in *Targets and Backgrounds XII: Characterization and Representation, Proceedings of SPIE*, Orlando, FL, USA, 2006, pp. 623908-9.

[17] E. F. Greneker III, J. L. Geisheimer, and D. Asbell, "Extraction of micro-Doppler data from vehicle targets at x-band frequencies," in *Radar Sensor Technology VI*, Orlando, FL, USA, 2001, pp. 1-9.

[18] E. F. Greneker III and V. B. Sylvester, "Use of the envelope detection method to detect micro-Doppler," in *Passive Millimeter-Wave Imaging Technology VI and Radar Sensor Technology VII*, Orlando, FL, USA, 2003, pp. 167-174.

[19] J. Fang, H. Meng, H. Zhang, and X. A. W. X. Wang, "A low-cost vehicle detection and classification system based on unmodulated continuous-wave radar," in *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*, 2007, pp. 715-720.

[20] S. L. Marple, Jr., "Large dynamic range time-frequency signal analysis with application to helicopter Doppler radar data," in *Signal Processing and its Applications, Sixth International Symposium on*, 2001, pp. 260-263.

[21] I. Bilik, J. Tabrikian, and A. Cohen, "GMM-based target classification for ground surveillance Doppler radar," *Aerospace and Electronic Systems, IEEE Transactions on,* vol. 42, pp. 267-278, January 2006.

[22] M. Otero, "Application of a continuous wave radar for human gait recognition," in *Signal Processing, Sensor Fusion, and Target Recognition XIV, Proceedings of SPIE*, Orlando, FL, USA, 2005, pp. 538-548.

[23] G. E. Smith, K. Woodbridge, and C. J. Baker, "Micro-Doppler signature classification," in *Radar, 2006. CIE '06. International Conference on*, 2006, pp. 1-4.

[24] Y. Yinan, L. Jiajin, Z. Wenxue, and L. Chao, "Target classification and pattern recognition using micro-Doppler radar signatures," in *Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, Seventh ACIS International Conference on*, 2006, pp. 213-217.

[25] M. Jahangir, K. M. Ponting, and J. W. O'Loghlen, "Robust Doppler classification technique based on hidden Markov models," *Radar, Sonar and Navigation, IEE Proceedings,* vol. 150, pp. 33-36, February 2003.

[26] J. W. Pitton, L. E. Atlas, and P. J. Loughlin, "Applications of positive time-frequency distributions to speech processing," *Speech and Audio Processing, IEEE Transactions on,* vol. 2, pp. 554-566, October 1994.

[27] V. C. Chen and H. Ling, *Time-frequency Transforms for Radar Imaging and Signal Analysis*. Boston, MA: Artech House, 2002.

[28] K. Harman and B. Hodgins, "The next generation of GUIDAR technology," in *Security Technology, 38th Annual International Carnahan Conference on*, 2004, pp. 169-176.

[29] M. Maki, C. Hill, and C. R. Malone, "User performance testing of the Perimitrax buried cable sensor," in *Security Technology, IEEE 33rd Annual International Carnahan Conference on*, Madrid, Spain, 1999, pp. 112-119.

[30] G. Hellard, "GRYFFIN TALOS taut wire perimeter detection system," in *Security Technology, 32nd Annual International Carnahan Conference on*, Alexandria, VA, 1998, pp. 206-209.

[31] Z. Zhang, P. Pouliquen, A. Waxman, and A. G. Andreou, "Acoustic micro-Doppler gait signatures of humans and animals," in *Information Sciences and Systems, CISS '07. 41st Annual Conference on*, Baltimore, MD, 2007, pp. 627-630.

[32] A. M. Sabatini and V. Colla, "A method for sonar based recognition of walking people," *Robotics and Autonomous Systems,* vol. 25, pp. 117-126, March 1998.

[33] R. Cutler and L. S. Davis, "Robust real-time periodic motion detection, analysis, and applications," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 22, pp. 781-796, August 2000.

[34] P. S. Huang, "Automatic gait recognition via statistical approaches for extended template features," *Systems, Man and Cybernetics, Part B, IEEE Transactions on,* vol. 31, pp. 818-824, October 2001.

[35] J. Little and J. Boyd, "Recognizing people by their gait: the shape of motion," *Videre: Journal of Computer Vision Research,* vol. 1, pp. 1-32, April 1998.

[36] H. Murase and R. Sakai, "Moving object recognition in eigenspace representation: gait analysis and lip reading," *Pattern Recognition Letters,* vol. 17, pp. 155-162, February 1996.

[37] F. A. Sadjadi and C. S. L. Chun, "Passive polarimetric IR target classification," *Aerospace and Electronic Systems, IEEE Transactions on,* vol. 37, pp. 740-751, April 2001.

[38] R. L. Rogers, D. L. Fisher, and H. D. Foltz, "Development and tests of a low cost passive millimeter wave sensor," Applied Research Laboratories, The University of Texas at Austin, Austin, TX, Technical Report: ARL-TR-97-3, April 15, 1997.

[39] G. N. Sinclair, R. N. Anderton, and R. Appleby, "Outdoor passive millimetre wave security screening," in *Security Technology, IEEE 35th International Carnahan Conference on*, London, UK, 2001, pp. 172-179.

[40] D. R. Wehner, *High-Resolution Radar*, 2nd ed. Boston, MA: Artech House, 1995.

[41] A. Zyweck and R. E. Bogner, "Radar target classification of commercial aircraft," *Aerospace and Electronic Systems, IEEE Transactions on,* vol. 32, pp. 598-606, April 1996.

[42] M. G. Anderson, "Multiple frequency continuous wave radar design for micro-Doppler extraction," Masters Thesis, Master of Science in Engineering (M.S.E.), The University of Texas at Austin, Austin, TX, 2005.

[43] M. I. Skolnik, *Introduction to Radar Systems*, 1st ed. New York: McGraw-Hill, 1962.

[44] S. O. Rice, "Mathematical analysis of random noise," *Bell Systems Tech. J.,* vol. 24, pp. 46-156, 1945.

[45] G. Brooker, "Detection of signals in noise," http://www.acfr.usyd.edu.au/teaching/4th-year/mech4721-signals/material/lecture%20notes/index.html, 2005

[46] X. Xiang-Gen, "A quantitative analysis of SNR in the short-time Fourier transform domain for multicomponent signals," *Signal Processing, IEEE Transactions on,* vol. 46, pp. 200-203, January 1998.

[47] X. G. Xia, G. Wang, and V. C. Chen, "Quantitative SNR analysis for ISAR imaging using joint time-frequency analysis-Short time Fourier transform," *Aerospace and Electronic Systems, IEEE Transactions on,* vol. 38, pp. 649-659, April 2002.

[48] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed. New York: Cambridge University Press, 1992.

[49] O. Marichev and M. Trott, "Inverse beta regularized," http://functions.wolfram.com/GammaBetaErf/InverseBetaRegularized/, 2005

[50] L. R. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: PTR Prentice Hall, 1993.

[51] S. Furui, "Cepstral analysis technique for automatic speaker verification," *Acoustics, Speech, and Signal Processing, IEEE Transactions on,* vol. 29, pp. 254-272, April 1981.

[52] F. K. Soong and A. E. Rosenberg, "On the use of instantaneous and transitional spectral information in speaker recognition," *Acoustics, Speech, and Signal Processing, IEEE Transactions on,* vol. 36, pp. 871-879, June 1988.

[53] M. Nosratighods, E. Ambikairajah, and J. Epps, "Speaker verification using a novel set of dynamic features," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, 2006, pp. 266-269.

[54] T. M. Cover and J. M. Van Campenhout, "On the possible orderings in the measurement selection problem," *Systems, Man and Cybernetics, IEEE Transactions on,* vol. 7, pp. 657-661, September 1977.

[55] P. M. Narendra and K. Fukunaga, "A branch and bound algorithm for feature subset selection," *Computers, IEEE Transactions on,* vol. C-26, pp. 917-922, September 1977.

[56] B. I. N. Yu and B. Yuan, "A more efficient branch and bound algorithm for feature selection," *Pattern Recognition,* vol. 26, pp. 883-889, 1993.

[57] H. Liu and H. Motoda, *Feature Selection for Knowledge Discovery and Data Mining*. Boston: Kluwer Academic Publishers, 1998.

[58] J. Doak, "An evaluation of feature-selection methods and their application to computer security," Technical Report CSE-92-18, University of California at Davis, Department of Computer Science,.

[59] L. Huan and Y. Lei, "Toward integrating feature selection algorithms for classification and clustering," *Knowledge and Data Engineering, IEEE Transactions on,* vol. 17, pp. 491-502, April 2005.

[60] K. Mineichi and S. Jack, "Comparison of algorithms that select features for pattern classifiers," *Pattern Recognition,* vol. 33, pp. 25-41, May 2000.

[61] P. Pudil, F. J. Ferri, J. Novovicova, and J. Kittler, "Floating search methods for feature selection with nonmonotonic criterion functions," in *Pattern Recognition, 1994. vol. 2 - Conference B: Computer Vision & Image Processing., Proceedings of the 12th IAPR International. Conference on*, 1994, pp. 279-283.

[62] Y. L. Chang, J. P. Fang, J. N. Liu, H. Ren, and W. Y. Liang, "A simulated annealing feature extraction approach for hyperspectral images," in *Geoscience and Remote Sensing Symposium, 2007. IGARSS 2007. IEEE International*, 2007, pp. 3190-3193.

[63] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley-Interscience, 2000.

[64] W. Siedlecki and J. Sklansky, "A note on genetic algorithms for large-scale feature selection," *Pattern Recognition Letters,* vol. 10, pp. 335-347, November 1989.

[65] J. S. Lee, "Hybrid genetic algorithms for feature selection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 26, pp. 1424-1438, November 2004.

[66] A. Jain and D. Zongker, "Feature selection: evaluation, application, and small sample performance," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 19, pp. 153-158, February 1997.

[67] H. Vafaie and K. De Jong, "Robust feature selection algorithms," in *Tools with Artificial Intelligence, 1993. TAI '93. Proceedings., Fifth International Conference on*, 1993, pp. 356-363.

[68] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research,* vol. 3, pp. 1157-1182, 2003.

[69] D. Ourston, S. Matzner, W. Stump, and B. Hopkins, "Applications of hidden Markov models to detecting multi-stage network attacks," in *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*, Big Island, Hawaii, 2003.

[70] J. W. Barnes, *Statistical Analysis for Engineers and Scientists: A Computer Based Approach*. New York: McGraw-Hill, 1994.

[71] R. Hubbard and J. S. Armstrong, "Why we don't really know what statistical significance means: a major educational failure," *Journal of Marketing Education,* vol. 28, pp. 114-120, August 2006.

[72] R. Fisher, "Statistical methods and scientific induction," *Journal of the Royal Statistical Society. Series B (Methodological),* vol. 17, pp. 69-78, 1955.

[73] V. N. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.

[74] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery,* vol. 2, pp. 121-167, June 1998.

[75] B. Schölkopf, C. J. C. Burges, and A. J. Smola, *Advances in Kernel Methods: Support Vector Learning*. Cambridge, MA: MIT Press, 1999.

[76] T. Joachims, "Text categorization with suport vector machines: learning with many relevant features," in *Machine Learning, Proceedings of the 10th European Conference on*, 1998, pp. 137-142.

[77] E. Osuna, R. Freund, and F. Girosit, "Training support vector machines: an application to face detection," in *Computer Vision and Pattern Recognition, Proceedings of the IEEE Computer Society Conference on*, 1997, pp. 130-136.

[78] J. Liu, J. Gough, and B. Rost, "Distinguishing protein-coding from non-coding RNAs through support vector machines," *PLoS Genet,* vol. 2, p. e29, April 2006.

[79] C. J. C. Burges, "Simplified support vector decision rules," in *International Conference on Machine Learning*, 1996, pp. 71-77.

[80] K. Crammer and Y. Singer, "On the learnability and design of output codes for multiclass problems," *Machine Learning,* vol. 47, pp. 201-233, May 2002.

[81] H. Chih-Wei and L. Chih-Jen, "A comparison of methods for multiclass support vector machines," *Neural Networks, IEEE Transactions on,* vol. 13, pp. 415-425, March 2002.

[82] D. A. Reynolds, "A Gaussian mixture modeling approach to text-independent speaker identification," Ph.D. Thesis, Doctor of Philosophy in Electrical Engineering, Georgia Institute of Technology, Atlanta, GA, 1992.

[83] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing,* vol. 10, pp. 19-41, January 2000.

[84] R. C. Rose, E. M. Hofstetter, and D. A. Reynolds, "Integrated models of signal and background with application to speaker identification in noise," *Speech and Audio Processing, IEEE Transactions on,* vol. 2, pp. 245-257, April 1994.

[85] J. J. Verbeek, "Mixture models for clustering and dimension reduction," Ph.D. Dissertation, University of Amsterdam, The Netherlands, 2004.

[86] K. Markov and S. Nakagawa, "Frame level likelihood normalization for text-independent speaker identification using Gaussian mixture models," in *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, 1996, pp. 1764-1767.

[87] J. J. Verbeek, N. Vlassis, and B. KrÖse, "Efficient greedy learning of gaussian mixture models," *Neural computation,* vol. 15, pp. 469-485, 2003.

[88] A. K. Jain, R. P. W. Duin, and M. Jianchang, "Statistical pattern recognition: a review," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 22, pp. 4-37, January 2000.

[89] A. Flexer, "Statistical evaluation of neural network experiments: Minimum requirements and current practice," in *Proceedings of the 13th European Meeting on Cybernetics and Systems Research*. vol. 2, 1996, pp. 1005–1008.

Vita

Michael Glen Anderson was born in Waterloo, Iowa on October 9th, 1979, the son of Alfrieda Helen Anderson and Glen Melvin Anderson. Michael attended Judson Senior High School in Converse, Texas and graduated in May 1998. He began his collegiate studies at The University of Texas at Austin in the Fall of 1998. Michael held internships at the Armstrong Research Laboratory at Brooks Air Force Base (now Brooks City-Base), Southwest Research Institute, and Applied Research Laboratories. Michael received the degrees of Bachelor of Science in Electrical Engineering and Master of Science in Engineering, both from The University of Texas at Austin, in May 2002 and May 2005, respectively. Following the award of the M.S.E., Michael accepted a full-time staff position at Applied Research Laboratories in Austin, Texas and also began work in pursuit of the Ph.D. degree.

Publications:

M. G. Anderson and R. L. Rogers, "Micro-Doppler analysis of multiple frequency continuous wave radar signatures," in *Radar Sensor Technology XI, Proceedings of SPIE*, vol. 6547, Orlando, FL, USA, 2007, pp. 65470A-10.

Permanent address: 13634 Merseyside Dr.
Pflugerville, TX 78660

This dissertation was typed by the author.